

2/1/74

DIGITAL COMMUNICATIONS STUDY  
FINAL REPORT

January 1, 1972 - January 1, 1973

Prepared for

National Aeronautics and Space Administrations

Goddard Space Flight Center

Under

NASA GRANT NGR - 33-006-020

PIBEE/EP - 73 - 006

Robert R. Boorstyn, Principal Investigator

DEPARTMENT OF  
ELECTRICAL ENGINEERING AND ELECTROPHYSICS

POLYTECHNIC INSTITUTE OF BROOKLYN

N74-21833

Unclas  
G3/08 37061

(NASA-CR-138129) DIGITAL COMMUNICATIONS  
STUDY Final Report, 1 Jan. 1972 - 1  
Jan. 1973 (Polytechnic Inst. of Brooklyn)  
102 p HC \$8.25 CSCI 09B

Reproduced by  
NATIONAL TECHNICAL  
INFORMATION SERVICE  
US Department of Commerce  
Springfield, VA. 22151

## N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM THE BEST COPY FURNISHED US BY THE SPONSORING AGENCY. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE.

DIGITAL COMMUNICATIONS STUDY  
FINAL REPORT

January 1, 1972 - January 1, 1973

Prepared for

National Aeronautics and Space Administrations

Goddard Space Flight Center

Under

NASA GRANT NGR - 33-006-020

PIBEE - 73 - ...

Robert R. Boorstyn, Principal Investigator

DEPARTMENT OF  
ELECTRICAL ENGINEERING AND ELECTROPHYSICS  
POLYTECHNIC INSTITUTE OF BROOKLYN

TABLE OF CONTENTS

	<u>Section</u>
Introduction	0
Signal Processing with Finite State Machines	I
Signal Parameter Estimation from Discrete- Time Observation	II
Digital Filtering for Radar Signal Processing Applications	III
On the Output of Multiple Server Queues Where Not All Servers Are Identical	IV

PRECEDING PAGE BLANK NOT FILMED

## 0. INTRODUCTION

This final report summarizes most of the research sponsored by the National Aeronautics and Space Administration under Grant NGR - 33-006-020 during the period January 1, 1972 to January 1, 1973. The research supported by this grant deals mostly with problems of digital data transmission and includes some new work on computer communication networks. There are four self-contained sections labeled I through IV. Each section has its own figures, references and equation numbering.

Section I, Signal Processing with Finite State Machines, continues the work on finite memory detection. New results for a time-invariant machine are given, optimum time-varying machines for detection are presented, and the structure and performance of these machines are developed. Problems of practical significance are discussed. Many talks have been given on this subject. One paper has been accepted for publication, and a second is almost completed. Invited papers were presented at the IEEE Convention, in New York in March 1972 and at the Southeastern Systems Symposium in North Carolina in March 1973. These papers have been published in the records of the symposia. Two papers were presented at the International Symposium on Information Theory in California in January 1972. This work was performed by R. R. Boorstyn, P. F. Lynn, and R. W. Muiise and is continuing.

Section II discusses work on Signal Parameter Estimation from Discrete-Time Observations. New results relevant to the problem of estimating several single-frequency tones from a finite number of noisy, discrete-time observations are presented. This problem has application to data systems testing, radar, and other measurements. A paper will be

presented at the International Symposium on Information Theory in Israel in June 1973 and will be submitted for publication. This work was performed by R. R. Boorstyn and D. C. Rife.

Section III contains work on Digital Filtering for Radar Signal Processing Applications. A novel approach in synthesizing digital filters for signal processing applications is presented. This synthesis method takes advantage of the known signal waveform structure and results in many fewer digital computations as compared to convolution processing. This approach is particularly suited to synthesis of matched filters for radar signal processing and yields matched or approximately matched filters which simultaneously have very low storage and very low computational requirements. A paper has been published in the Transactions on Audio and Electronics, IEEE, March 1972. This work was performed by R. R. Boorstyn and J. D. Echard.

Section IV contains work On Multiple Server Queues Where Not All Servers are Identical. This is an attempt to derive some properties of networks of queues by considering the individual outputs of multiple server queues. It is shown under what conditions the outputs retain the Poisson character of the input. This work was performed by R. R. Boorstyn and P. McGregor.

The PIB faculty participating in this program were R. R. Boorstyn, who prepared the final report, R. A. Haddad, M. Schwartz, and J. K. Wolf.

## I. SIGNAL PROCESSING WITH FINITE STATE MACHINES

This paper is concerned with the implementation of communications receivers using digital processors or digital computers and the effect of this implementation on performance. One can consider conventional (analog) receivers to consist of filters, summers, multipliers, comparators, matched filters, phase-locked loops, etc. The usual method in converting these receivers into digital devices is to replace each analog component with its digital counterpart. Thus, after the required sampling, digital filters replace the original analog filters, summers and multipliers are replaced by digital summers and multipliers, etc. If the sampling is fast enough there should be little deleterious effect on performance.

However, there is one further constraint that is the dominant concern of this paper. Every coefficient, every data sample, and the results of every operation must be represented and stored by a finite length word, i. e., a finite number of bits. For a standard computer approximately 30 bits are available for each word and this should be sufficient. For special purpose computers, or in time-sharing operation where the device is to be used simultaneously for many similar operations, it is important to investigate the effect of reducing the storage capability considerably. It is clear that as the number of bits is reduced, the performance will ultimately deteriorate. If the amount of storage is increased from this point, the performance will be reasonably satisfactory. However, if the size of storage is reduced further the question arises: is there an entirely new digital structure that will perform well with a minimum amount of storage? This question is answered affirmatively in this paper.

To illustrate, consider a simple binary receiver. One of two equally

likely signals are to be sent, and  $n$  independent samples  $X_1, X_2, \dots, X_n$  are received. If one of the signals (denoted by hypothesis  $H_0$ ) is sent then each of the  $X_k$  has density  $f_0(x)$ . Otherwise ( $H_1$ ) they have density  $f_1(x)$ . The optimum receiver forms

$$(1) \quad \lambda(k) = \ell \ln \frac{f_1(X_k)}{f_0(X_k)} + \lambda(k-1), \quad k = 1, \dots, n,$$

where  $\lambda(0) = 0$ , and decides in favor of  $H_1$  if  $\lambda(n) > 0$ . This is just a recursive version of the likelihood ratio.

In general all the  $n$  data samples must be remembered with infinite precision to make the optimum decision. If the data is assumed to be collected sequentially in time, then equation (1) states that after  $k$  samples are received these first  $k$  samples of data need effectively only be "remembered" by storing the single number  $\lambda(k)$ . Then  $\lambda(n)$  is used to make the decision. But the  $\lambda(k)$  must still be stored with infinite precision. It can be shown that, in simple examples, merely uniformly quantizing each  $\lambda(k)$  to a finite number of bits and using equation(1) results in a deterioration of performance if the number of samples is roughly greater than the number of levels of quantization.

There are, however, receiver structures which incorporate fundamentally this finite storage constraint, here called "finite memory", and perform almost as well as the optimum infinite memory receiver given by equation (1). These receivers can be modelled as finite state machines. After the  $(k-1)^{\text{st}}$  sample the machine may be in one of  $m$  states, e. g., the  $m$  levels of a  $\log_2 m$  bit quantizer. This state at time  $k-1$ ,  $S_{k-1}$ , and the next sample  $X_k$  are used to determine the state at time  $k$ . The final decision is based upon the state at time  $n$ . An example of such a finite



machine suitable for detection is shown in Figure 1. This is essentially a polarity counter with a null-zone and overflow.

This machine, which can be called a "linear" machine, operates as follows. If any sample, say  $X_k > d$ , then the state of the machine moves one step to the right. If  $X_k < -d$  it moves one step to the left. Otherwise it remains in the same state. It remains in the end states unless, e. g.,  $X_k < -d$  in state  $m$ . The machine initially starts in one of the middle two states with equal probability (assuming  $m$  is even and symmetrical statistics). If at time  $n$ ,  $S_n > \frac{m}{2}$ ,  $H_1$  is decided. Otherwise  $H_0$  is decided. This is a time-invariant machine since the transition rules from state to state do not change with  $k$ . Cover [ 1 ] has shown that, remarkably, as the length of the data sequence becomes infinite the probability of error  $P_e(m, n)$  asymptotically approaches zero if  $m \geq 4$  for time-varying machines. Indeed, this also holds true for  $m \geq 2$  for certain statistics. Hellman and Cover [ 2 ] have found the asymptotic probability of error for time-invariant machines and have shown that this also goes to zero under certain conditions. They have also shown that the optimum machines resemble, asymptotically, that of Figure 1 but with  $d \rightarrow \infty$ . The convergence of probability of error is slow and little insight is gained from this asymptotic behavior about optimum or good machines and their performance for finite data sequences.

Lynn and Boorstyn [ 3 ] have evaluated the performance of the "linear" machine for finite values of  $m$  and  $n$ . Typical results are summarized in Figures 2 and 3. Briefly these indicate that for 3 bits of memory ( $m=8$ ) performance close to optimum (infinite memory) can sometimes be achieved. These numerical results are for Gaussian statistics:  $f_1(x)$  Gaussian with mean  $\mu > 0$  and variance  $\sigma^2$ ,  $f_0(x)$  Gaussian with mean  $-\mu$  and variance  $\sigma^2$ .

It is comforting that for this "linear" machine an explicit and simple expression for the probability of error can be obtained. This is given below in equation 2.

$$(2) \quad P_e^L(m, n) = \frac{1}{1 + (a_3/a_1)^{m/2}} + \frac{(a_3 - a_1)^{m-1}}{m} \sum_{\substack{j=1 \\ (j \text{ odd})}}^{m-1} \frac{\beta_j^n}{1 - \beta_j}$$

where  $\beta_j = 2\sqrt{a_1 a_3} \cos(\pi j/m) + a_2$

$m$  is even

$$a_1 = \Pr\{X_k < -d \mid H_1\}$$

$$a_2 = \Pr\{|X_k| \leq d \mid H_1\}$$

$$a_3 = \Pr\{X_k > d \mid H_1\}$$

$$a_1 + a_2 + a_3 = 1$$

$$\text{and } a_3 > a_1.$$

The first term in (2) is the steady state term since the second term vanishes as  $n \rightarrow \infty$ . For  $a_1 < a_3 < 1$   $\beta_j$  can be tightly bounded by  $\beta = a_2 \pm 2\sqrt{a_1 a_3}$  for all  $j$ . Thus the second term can be bounded by  $\frac{1}{2}(a_3 - a_1)\beta^n/(1 - \beta)$  which is independent of  $m$ . Thus  $m$  appears only in the first term and  $n$  in the second term. Both expressions are functions of  $a_i$  and through them of  $d/\sigma$  and  $\mu/\sigma$  for Gaussian statistics. Thus (2) should yield considerable insight about the setting of the threshold  $d/\sigma$  and the dependence upon signal-to-noise ratio  $\mu/\sigma$  as well as the nature of the variation of  $P_e$  with  $m$  and  $n$ . Some results for a four state machine are given in Figure 4.

To gain further insight we considered a machine with a different structure which we call a majority rule machine. This looks only at the last  $2\ell - 1$  non-null ( $|X_k| > d$ ) samples and decides by majority rule. E. g., if more than half of them are  $> d$  then  $H_1$  is chosen. This machine has  $2^{2\ell - 1}$

states of memory and its performance is relatively easy to evaluate.

This machine performs about as well as a linear machine with logarithmically fewer ( $2\ell$ ) states but can serve as upper and lower bounds for the linear machine. (The lower bound may not always be applicable.) Specifically if the superscript L denotes the linear machine and M, the majority rule machine, then

$$(3) \quad \text{Pe}^M(2^{2\ell+1}, n) \leq \text{Pe}^L(2\ell, n) \leq \text{Pe}^M(2^{2\ell-1}, n).$$

A comparison of these two machines is given in Figure 5. Neither of these machines are optimum, although Hellman has shown that the linear machine for  $m=2$  is optimum.

Improved performance can be obtained if the machines are allowed to be time-varying. Muise and Boorstyn [4] have found the optimum time-vary-machine. It can be described as follows. If in state  $i$  at time  $k-1$  go to state  $j$  after receiving  $X_k$  if

$$(4) \quad \gamma_j(k) < \ln \frac{f_1(X_k)}{f_0(X_k)} + L_i(k-1) \leq \gamma_{j+1}(k)$$

$L_i(k-1)$  is defined to be

$$(5) \quad L_i(k-1) = \ln \frac{\Pr\{S_{k-1}=i | H_1\}}{\Pr\{S_{k-1}=i | H_0\}}$$

and can be viewed as a likelihood ratio where the "data" is the only observation that the machine can make, i. e., its state, at time  $k-1$ . Equation (4) can be viewed as a time-varying quantizer where the  $\gamma_j(k)$  are the threshold values and the  $L_i(k-1)$  are the representation values. If, at time  $n$ , the machine is in a state  $S_i$  for which  $L_i(n) > 0$  then  $H_1$  is decided; otherwise  $H_0$ .

Algorithms have been obtained for calculating the sets of coefficients

$L_1(k-1)$  and  $\gamma_j(k)$  and for evaluating the performance of these optimum machines. The threshold values are given by

$$(6) \quad \gamma_j(k) = \frac{f_j(k) - f_{j-1}(k)}{e_j(k) - e_{j-1}(k)}, \quad j=2, \dots, m$$

where

$$e_j(k) = P \{ \text{deciding } H_1 \text{ at time } n | S_k=j, H_1 \}$$

$$f_j(k) = P \{ \text{deciding } H_1 \text{ at time } n | S_k=j, H_0 \}$$

It can be seen that if at time  $k$  the design of the machine in the past is known then the coefficients in (5) can be evaluated. However, the coefficients in (6) depend upon the future design of the machine. The algorithm essentially starts with an initial design and iteratively runs back and forth in time applying equations (6) and (5) in turn. It has been proven that this algorithm converges and for typical examples the convergence is rapid. Figure 6 displays the coefficients for an example. Note that the coefficients for  $n=8$  are not simply extensions of those for  $n=4$  although the difference does not seem to be great. This suggests a suboptimum and simpler design which would find the best coefficients for each  $k$  as if it were the time of decision.

Typical results for the optimum machine are also shown in Figures 2 and 3 for Gaussian statistics. Also given there for reference is the performance of the optimum infinite memory machine and of the polarity counter. These results show that remarkably good performance can be obtained with relatively simple machines and a very small memory-size limitation.

The basic results in the optimization above can be inferred from the

following argument [ 5] . Let C denote a correct decision at time n. Then at time k

$$(7) \quad P(C | S_{k-1}=i) = \sum_{j=0}^1 \sum_{\ell=1}^m \int_{-\infty}^{\infty} \Pr(C | S_k=\ell, S_{k-1}=i, x_k, H_j) \cdot \\ \Pr(S_k=\ell | S_{k-1}=i, x_k, H_j) \cdot \\ f_{x_k}(x_k | S_{k-1}=i, H_j) \Pr(H_j | S_{k-1}=i) dx_k$$

But the first term in (7) is  $\Pr(C | S_k=\ell, H_j)$  and is used to form the terms in (6), the second term is  $\Pr(S_k=\ell | S_{k-1}=i, x_k) = 0$  or 1 depending on the decision rule at time k, the third term is  $f_{x_k}(x_k | H_j) = f_j(x_k)$ , and the fourth term is used to form (5). If  $A_{i\ell}(k)$  is the region of  $x_k$  such that a transition will occur from  $S_{k-1}=i$  to  $S_k=\ell$ , then

$$(8) \quad P(C | S_{k-1}=i) = \sum_{\ell=1}^m \int_{A_{i\ell}(k)} \left[ \sum_{j=0}^1 \Pr(C | S_k=\ell, H_j) f_j(x_k) \cdot \Pr(H_j | S_{k-1}=i) \right] dx_k$$

The optimum transition rule for going from  $S_{k-1}$  to  $S_k$  can be found from (8) by, for each i and  $x_k$ , finding the  $\ell$  that maximizes the bracketed expression. Muise and Boorstyn, in addition to other results, have given structure to this updating rule and proven its optimality. From (8) it is easy to extend this results to include time-varying statistics, many hypotheses, and time-varying memory sizes.

Although the memory has been constrained to be finite by limiting it to a sequence of states from a finite alphabet, the coefficients in (4) in the above analysis are not so constrained. (The arithmetic operation indicated in (4) is performed instantaneously, i. e., not stored, so that no discrete limitation need be put upon it. ) These coefficients must be stored and there

are approximately 2nm of them! It is not known at this time what the effect would be if the storage of these coefficients were also limited. However the storage of the states is data sensitive (read-write memory) while the coefficients are (except for an adaptive machine) fixed (read-only memory). These differences can be exploited, especially by time sharing the coefficients and the arithmetic operation with many similar detection processes. A sketch of such a structure is given in Figure 7.

Further research is continuing on many aspects of this problem - optimum time-invariant machines, algorithms for M-ary detection, other communications receivers, such as equalizers, more practical transition rule algorithms, time-sharing implementation, etc. It is hoped that these investigations will provide a new and more efficient viewpoint for designing communications systems incorporating fundamental digital constraints.

#### REFERENCES

1. Cover, T.M., "Hypothesis Testing with Finite Statistics." Ann. Math. Stat., vol. 40, 1969, pp. 828-835.
2. Hellman, M.E. and Cover, T.M., "Learning with Finite Memory." Ann. Math. Stat., vol. 41, 1970, pp. 765-782.
3. Lynn, P.F. and Boorstyn, R.R., "Time-Invariant Finite Memory Detectors," accepted for publication in Trans. Information Theory, IEEE.
4. Muise, R.W. and Boorstyn, R.R., "Optimum, Time-Varying, Finite Memory Detection," to be submitted for publication
5. Mullis, C.T., "A Class of Finite Memory Decision Processes," University of Colorado, 1968.

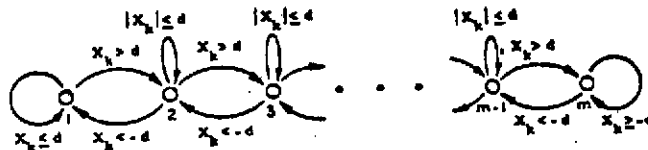
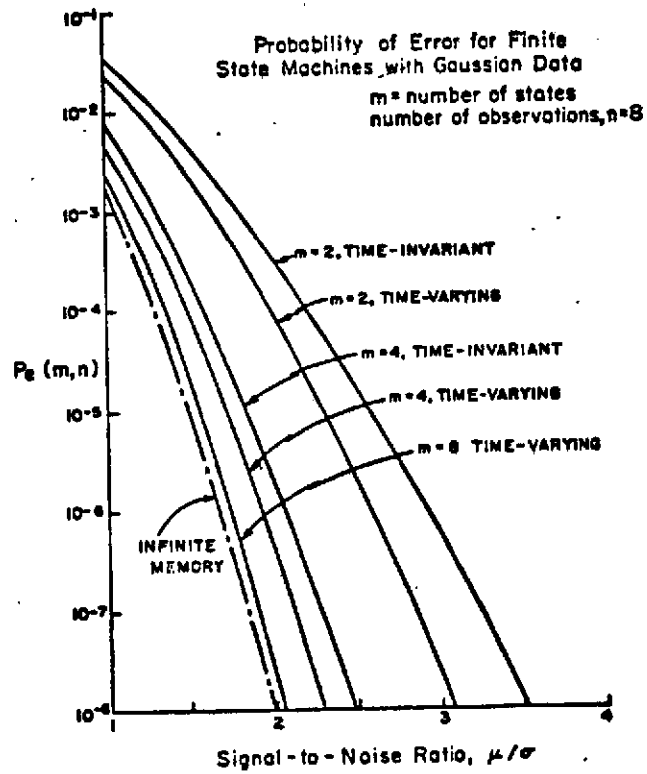


Figure 1. A "Linear" Machine



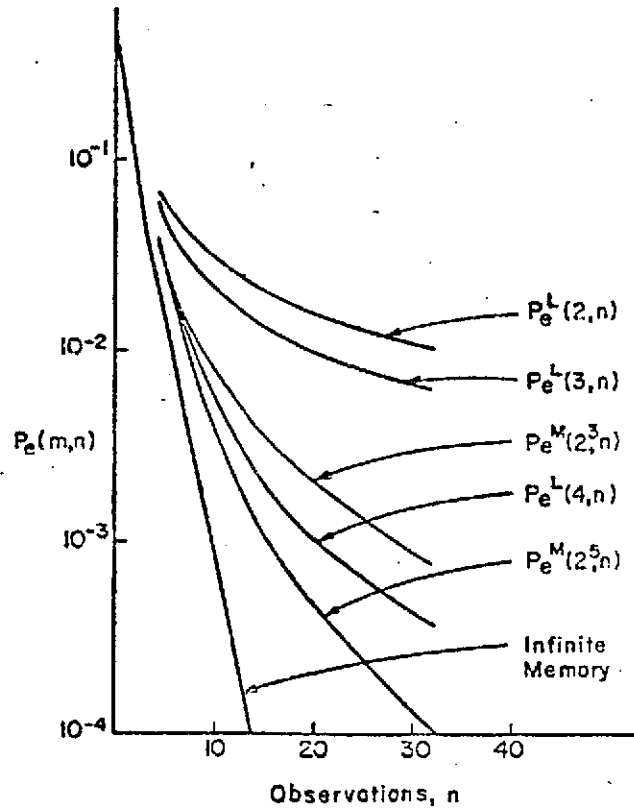
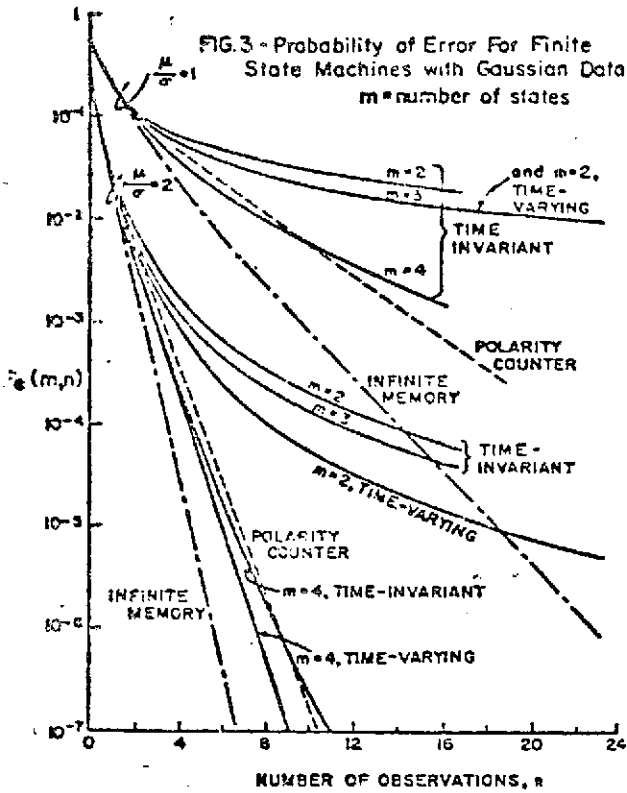


FIGURE 5. Comparison of Performance of the Linear Detector and the Majority Rule Detector for Gaussian Data,  $P = \mu/\sigma = 1$

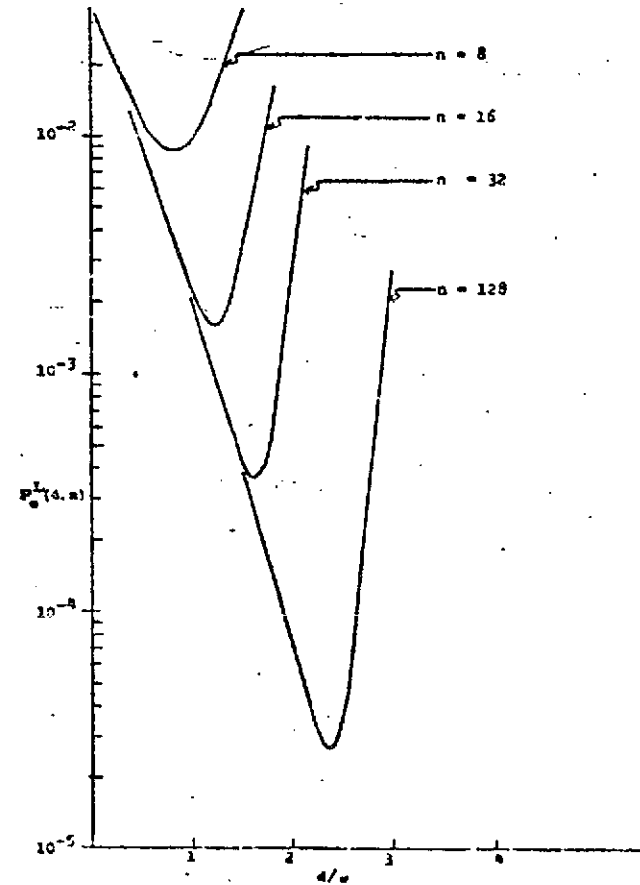


Figure 4. Prob. of Error vs. Normalized Threshold for a Four State Linear Detector for Gaussian Distributed Observations and Signal to Noise Ratio  $\rho = \sigma^2/\sigma^2 = 1$ .

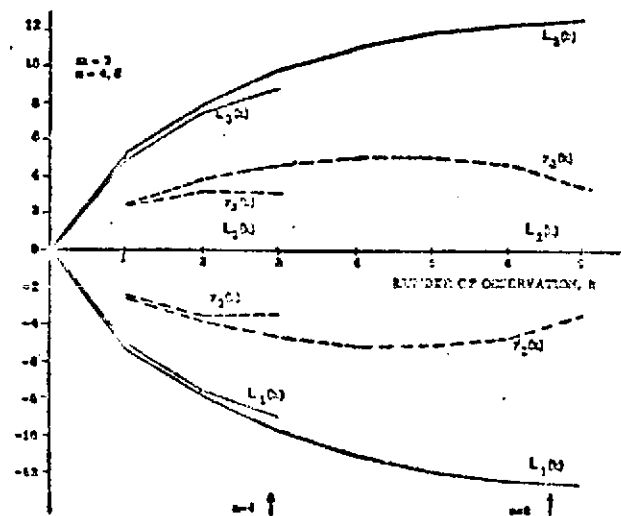


FIGURE 6. RECEIVER STRUCTURE FOR GAUSSIAN DATA ( $\mu/\sigma=2$ )



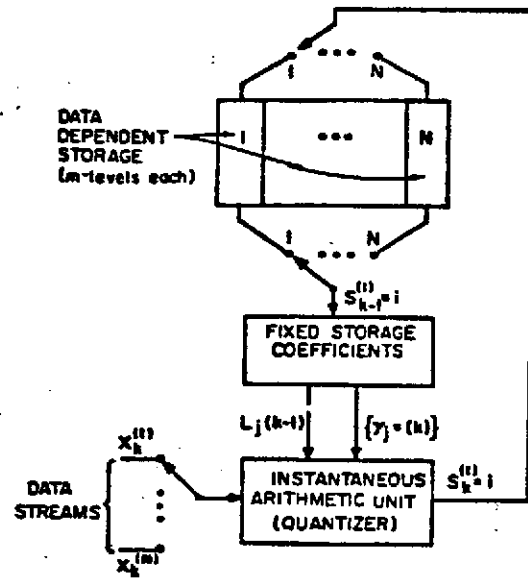


FIGURE 7: MACHINE CONFIGURATION -  
TIME SHARING MODE

## II. SIGNAL PARAMETER ESTIMATION FROM DISCRETE-TIME OBSERVATIONS

This paper is intended to present some new results relevant to the problem of estimating the parameters of several single-frequency tones from a finite number of noisy, discrete-time observations. This problem has application to data system testing, radar, and other measurement situations.

The general model is indicated on Figure 1. In general the signal has the form  $\sum_{i=1}^k b_i \exp [j(\omega_i t + \theta_i)]$ . In a working system the imaginary part may be derived from the real part by a Hilbert transformer or the imaginary part may not be processed at all.

Time does not permit an examination of all the possible cases. Therefore, we will only discuss the case of a single tone whose imaginary part is a single cycloidal tone. An understanding of this case is fundamental to an understanding of the other cases.

The parameter estimation problem was formulated and examined by Slepian in his often-quoted paper of 1954. [1] Our intent here is to study, in more detail, a specific variation of the problem.

The real part of the signal,  $s(t)$ , is  $b_0 \cos(\omega_0 t + \theta_0)$ . Suppose all three parameters are unknown, but only  $b_0$  and  $\omega_0$  are to be estimated. (The phase can be estimated but we will not do so.)

The computer input will be two sets of samples,

$$\underline{X} = [X_0, X_1, \dots, X_{N-1}]^T \text{ and } \underline{Y} = [Y_0, Y_1, \dots, Y_{N-1}]^T, \text{ where}$$

$$X_n = s(t_n) + W(t_n), \quad n = 0 \text{ to } N - 1, \quad (1)$$

$$Y_n = \underline{v}_s(t_n) + \underline{v}_W(t_n), \quad n = 0 \text{ to } N - 1, \quad (2)$$

and

$$s(t) = b_0 \sin(\omega_0 t + \theta_0). \quad (3)$$

$\check{W}(t)$  is the Hilbert transform of the noise,  $W(t)$ . We only consider the case of independent noise samples.

If we write  $\underline{Z} = \underline{X} + j\underline{Y}$  then the joint probability density function (p. d. f. ) of the elements of the sample vector  $\underline{Z}$  when the parameter vector is  $\underline{a}$  is given by

$$f(\underline{Z}; \underline{a}) = \left[ \frac{1}{\sigma^2 2\pi} \right]^N \exp \left\{ - \frac{1}{2\sigma^2} \sum_{n=0}^{N-1} \left[ (X_n - \mu_n)^2 + (Y_n - \nu_n)^2 \right] \right\} \quad (4)$$

where

$$\underline{a} = [\omega, b, \theta]^T, \quad (5)$$

$$\mu_n = b \cos(\omega t_n + \theta), \quad (6)$$

and

$$\nu_n = b \sin(\omega t_n + \theta). \quad (7)$$

We will write  $t_n = (n_0 + n)T$ .

In developing the topic we will consider three main aspects of the problem. First we will examine lower bounds to estimation error, in particular Cramér-Rao lower bounds. Then we will develop and analyze maximum-likelihood (ML) estimators of the signal parameters. Finally, we will discuss practical estimation algorithms and simulation results. The frequency estimation algorithm has a threshold effect, which we will also discuss.

### BOUNDS

Let us first look at lower bounds to estimation error. In an estimation (or measurement) system it is important to have numbers that indicate the best estimation that can be made with the available data (the observations).

In a measurement system RMS errors are important and are often used as a measure of system inaccuracy. Estimation bias is of secondary importance, although it is generally desirable to minimize bias. In this report we will generally find that the bias can be neglected. Thus RMS errors will be the important consideration.

Lower bounds to estimation accuracy have been studied by many people. Some of the better-known bounds are: the Cramer-Rao (C-R) bound [2] and its generalizations [3], the Bhattacharyya system of bounds [4], the Barankin bounds [5, 6], the Ziv-Zaki bounds [7], and the Chapman-Robbins bound [8]. The C-R bound is imbedded in the Bhattacharyya system of bounds and in the Barankin bounds. The Chapman-Robbins bound is related to the C-R bound. The difference is that the Chapman-Robbins bound avoids the need for the regularity conditions required in the C-R and Bhattacharyya bounds. Seidman has observed that many of the bounds are loose, especially at signal-to-noise ratios (SNR) above threshold [9]. Some are also difficult to use. We will use maximum-likelihood (ML) estimation and will generally be able to keep the bias very small. Thus, above threshold, the C-R bound will apply. We will separately evaluate threshold effects.

The generalized C-R bound is due to Rao [3]. It can be shown that this bound is the best (tightest) of a certain class of bounds [10].

Before reviewing the bound it is helpful to make some definitions. We assume the indicated operations are legitimate. (They are in this problem.)

Let  $H(\underline{Z}, \underline{a}) = \log f(\underline{Z}; \underline{a})$ . (9)

Let  $\underline{S}$  be the vector defined by its typical element:

$$S_i = \frac{\partial}{\partial a_i} H(\underline{Z}, \underline{a}) . \quad (10)$$

The Fisher information matrix,  $J(\underline{a})$ , is defined by

$$J = E \{ \underline{S} \underline{S}^T \} . \quad (11)$$

A typical element of  $J$  is

$$J_{ij} = E \{ H_{a_i} H_{a_j} \} = - E \{ H_{a_i a_j} \} \quad (12)$$

Let  $\hat{\underline{a}}(\underline{Z})$  be an estimator of  $\underline{a}$ . Let the matrix  $D$  be defined by typical element

$$D_{ij} = \frac{\partial}{\partial a_j} E \{ \hat{a}_i(\underline{Z}) \}. \quad (13)$$

Let the matrix  $C(\underline{a})$  be defined by

$$C(\underline{a}) = E \{ (\hat{\underline{a}} - \underline{a})(\hat{\underline{a}} - \underline{a})^T \}. \quad (14)$$

With these definitions and certain well-known regularity conditions on  $f(\underline{Z}; \underline{a})$  and  $J$ , the generalized C-R bound is the statement that the matrix  $C - DJ^{-1}D^T$  is positive semidefinite. From this we find that

$$\text{Var} \{ \hat{a}_i \} \geq [DJ^{-1}D^T]_{ii}, \quad i = 1, 2, 3, \dots, p, \quad (15)$$

where  $[ ]_{ii}$  denotes the  $i$ th diagonal element of  $[ ]$  and  $p$  is the number of elements in  $\underline{a}$ . We will ignore the conditions for equality to occur in (15) since they are generally not met in the problem at hand.

$$\text{If } \frac{\partial}{\partial a_i} E \{ \hat{a}_j \} = \delta_{ij}, \quad (16)$$

which happens when  $\hat{\underline{a}}$  is an unbiased estimator of  $\underline{a}$ , then  $D = I$  and the bounds become: [11]

$$\text{Var} \{ \hat{a}_i \} \geq J^{ii}, \quad i = 1 \text{ to } p. \quad (17)$$

It can be shown that with  $f(\underline{Z}; \underline{a})$  as given above

$$J_{ij} = \frac{1}{\sigma^2} \sum_{n=0}^{N-1} \left[ \frac{\partial \mu_n}{\partial a_i} \frac{\partial \mu_n}{\partial a_j} + \frac{\partial \nu_n}{\partial a_i} \frac{\partial \nu_n}{\partial a_j} \right] \bigg|_{\underline{a} = \underline{a}_0}. \quad (18)$$

where

$$\underline{a}_0 = [\omega_0, b_0, \theta_0]^T. \quad (19)$$

After determining the elements of the J matrix, the details of which are omitted, one obtains:

$$J = \frac{1}{\sigma^2} \begin{bmatrix} b_1^2 T^2 (n_0^2 N + 2n_0 P + Q) & 0 & b_1^2 T (n_0 N + P) \\ 0 & N & 0 \\ b_1^2 T (n_0 N + P) & 0 & b_1^2 N \end{bmatrix}, \quad (20)$$

where

$$P = \sum_{n=0}^{N-1} n = \frac{N(N-1)}{1}, \quad (21)$$

$$Q = \sum_{n=0}^{N-1} n^2 = \frac{N(N-1)(2N-1)}{6}, \quad (22)$$

and  $n_0 T$  is the time at which the first sample is taken.

From here on let us suppose  $n_0 = 0$ . This will be convenient later.

From (20) the bounds will depend upon  $n_0$ . We will not dwell upon this dependence.

We can see from the zeros in J that the level bound,  $\text{Var}\{\hat{a}_2\} \geq J^{22}$ , will be the same whether or not the phase and/or frequency are known. If the phase were known the J matrix would be

$$\begin{bmatrix} J_{11} & 0 \\ 0 & J_{22} \end{bmatrix},$$

for example. If the frequency were known the J matrix would be

$$\begin{bmatrix} J_{22} & 0 \\ 0 & J_{33} \end{bmatrix}.$$

Observe that the elements of  $J$  are not functions of  $\omega$  or  $\theta$ . This is not true in the more general cases.

Inversion of the  $J$  matrix is easy. When all of the parameters are unknown the results are:

$$\text{Var} \{ \hat{\omega}_0 \} \geq \frac{12\sigma^2}{b_0^2 T^2 N(N^2-1)} \quad (23)$$

$$\text{Var} \{ \hat{b}_0 \} \geq \sigma^2 / N \quad (24)$$

$$\text{Var} \{ \hat{\theta}_0 \} \geq \frac{2\sigma^2(2N-1)}{b_0^2 N(N+1)} \quad (25)$$

### MAXIMUM LIKELIHOOD ESTIMATION

Now let us look at ML estimation of  $\omega_0$  and  $b_0$ , the frequency and level. The ML estimate of  $\underline{a}$  is the value of  $\underline{a}$ , say  $\hat{\underline{a}}$ , that maximizes  $f(\underline{Z}; \underline{a})$  when  $\underline{Z}$  is the observed sample vector.

The maximum of  $f(\underline{Z}; \underline{a})$  will occur at the maximum of  $\log(f) = H(\underline{Z}, \underline{a})$ , or at the maximum of the function

$$L = -\frac{1}{N} \sum_n (X_n - \mu_n)^2 + (Y_n - \nu_n)^2. \quad (26)$$

Since  $\sum X_n^2$ ,  $\sum Y_n^2$ , and  $\sigma^2$  are not affected by  $\omega$ ,  $b$ , or  $\theta$ , we can drop them from  $L$  and use  $L_1$ :

$$L_1 = \frac{2}{N} \sum_n (X_n \mu_n + Y_n \nu_n) - \frac{1}{N} \sum_n \left[ \mu_n^2 + \nu_n^2 \right] \quad (27)$$

$$= \frac{2b}{N} \sum_n [X_n \cos(n\omega T + \theta) + Y_n \sin(n\omega T + \theta)] - b^2 \quad (28)$$

Define

$$F(\underline{Z}, \omega) = \sum_n (X_n \cos n\omega T + Y_n \sin n\omega T) \quad (29)$$

and

$$G(\underline{Z}, \omega) = \sum_n (Y_n \cos n\omega T - X_n \sin n\omega T). \quad (30)$$

Then

$$L_1 = \frac{2b}{N} [F(\underline{Z}, \omega) \cos \theta + G(\underline{Z}, \omega) \sin \theta] - b^2 \quad (31)$$

Assuming  $\hat{b}_0 > 0$ , we need to maximize  $F \cos \theta + G \sin \theta$  over both  $\theta$  and  $\omega$ . The maximum over  $\theta$  is  $\sqrt{F^2 + G^2}$ . Thus the ML estimate of  $\omega_0$  is the value of  $\omega$  that maximizes

$$B(\omega) = [F^2(\underline{Z}, \omega) + G^2(\underline{Z}, \omega)] / N^2 \quad (32)$$

$$\text{The ML estimate of } b_0 \text{ is } \hat{b}_0 = \sqrt{B(\hat{\omega})}. \quad (33)$$

#### Relationship to DFT

Recall that the discrete Fourier transform (DFT) of the vector  $\underline{Z}$  is the set of complex numbers: [12, 13, 14]

$$A_K = \frac{1}{N} \sum_{n=0}^{N-1} Z_n e^{-j2\pi nk/N}, \quad k = 0, 1, 2, \dots, N-1. \quad (34)$$

If we define a function  $A(\omega)$ :

$$A(\omega) = \frac{1}{N} \sum_{n=0}^{N-1} Z_n e^{-jn\omega T}, \quad (35)$$

then



$$A_k = A \left[ \frac{2\pi k}{NT} \right], \quad k = 0, 1, \dots, N-1. \quad (36)$$

It is easy to show that the function  $B(\omega)$  defined above is

$$B(\omega) = |A(\omega)|^2. \quad (37)$$

We see, therefore, that ML estimation of  $\omega_0$  and  $b_0$  is related to the DFT of the sample vector  $\underline{Z}$ . This fact will suggest a practical algorithm later.

#### Properties of $\hat{\omega}_0$

Because of the nonlinear nature of the estimation algorithm we are unable to derive the distribution functions for  $\hat{b}_0$  and  $\hat{\omega}_0$ . We can, however, show:

1. The p.d.f. of  $\hat{\omega}$  is symmetrical about  $\omega_0$ , modulo  $\omega_s$ .
2.  $\text{Var}\{\hat{\omega}\}$  is proportional to  $\omega_s^2$  and independent of  $\omega_0$ . ( $\omega_s = 2\pi/T$ .)

Noise Model. The following noise model leads to easy proofs. Let  $\{V_n\}$  be a set of independent Rayleigh r.v. with parameter 1. That is

$$f_{V_n}(v) = \begin{cases} v \exp[-v^2/2], & v \geq 0 \\ 0, & v < 0 \end{cases} \quad (38)$$

Let  $\{\phi_n\}$  be a set of independent r.v. uniformly distributed over  $(-\pi, \pi)$ .

We model the Gaussian samples as

$$W_n = \sigma V_n \cos \phi_n \quad (39)$$

and

$$\hat{W}_n = \sigma V_n \sin \phi_n. \quad (40)$$

Analysis . Recall

$$Z_n = X_n + jY_n .$$

Then, using our noise model,

$$Z_n = b_1 e^{j(n\omega_1 T + \theta_1)} + \sigma V_n e^{j\phi_n} . \quad (41)$$

Thus

$$\begin{aligned} A(\omega) &= \frac{1}{N} \sum_n Z_n e^{-jn\omega T} \\ &= \frac{e^{j\omega_1 T}}{N} \left[ \sum_n b_1 e^{-jn\beta} + \sigma V_n e^{-j(n\beta - \phi_n + \theta_1 + n\omega_1 T)} \right] , \end{aligned} \quad (42)$$

$$\text{where } \beta = (\omega - \omega_1)T . \quad (43)$$

$$\text{Let } \gamma_n = \phi_n - \theta_1 - n\omega_1 T , \quad n = 0 \text{ to } N - 1 . \quad (44)$$

Since the  $\phi_n$  are independent and uniform on  $(-\pi, \pi)$  , in effects so are the  $\gamma_n$  . We are therefore justified in writing

$$A(\omega) = \frac{e^{j\theta_0}}{N} \sum_n \left[ b_0 e^{-jn\beta} + \sigma V_n e^{-j(n\beta - \gamma_n)} \right] \quad (45)$$

$$\begin{aligned} \text{Thus } B(\omega) &= \frac{1}{N^2} \left[ b_0 \sum_n \cos n\beta + \sigma \sum_n V_n \cos(n\beta - \gamma_n) \right]^2 + \\ &\quad \frac{1}{N^2} \left[ b_0 \sum_n \sin n\beta + \sigma \sum_n V_n \sin(n\beta - \gamma_n) \right]^2 . \end{aligned} \quad (46)$$

The independence from  $\theta_0$  is now obvious.

Without loss of generality, let  $\hat{\beta}$  be the value of  $\beta$  in the range  $(-\pi, \pi)$  that maximizes  $B(\omega)$  . The estimator,  $\hat{\omega}_0$  , will then have the value:

$$\hat{\omega}_0 = \omega_0 + \frac{\omega_s \hat{\beta}}{2\pi} ; \text{ modulo } \omega_s . \quad (47)$$

Observe that  $B(\omega)$  is an even function of the pair  $\beta, \gamma$ .

(48)

The statistics of  $-\gamma$  are the same as the statistics of  $\gamma$ . Thus the statistics of  $-\beta$  must be the same as the statistics of  $\beta$ .

Hence the p.d.f. of  $\hat{\beta}$  must be an even function of  $\beta$  and  $E\{\hat{\beta}\} = 0$ . From (46), the statistics of  $\hat{\beta}$  do not depend upon  $\omega_0$  or  $\theta_0$ , but do depend upon the SNR,  $b_0/2\sigma^2$ .

Since we choose  $\hat{\omega}_0$  according to (47), the p.d.f. of  $\hat{\beta}$  is related to the p.d.f. of  $\hat{\omega}$  in the manner illustrated on Figure 2. The p.d.f. of  $\hat{\omega}_0$  is even about  $\omega_0$  except for the part from  $2\omega_0$  to  $\omega_s$ , when  $\omega_0 < \frac{\omega_s}{2}$  (or part from 0 to  $2\omega_0 - \omega_s$ , when  $\omega_0 > \frac{\omega_s}{2}$ ).

Consider the situation when  $\omega_0 < \frac{\omega_s}{2}$ . If  $\Pr\{2\omega_0 < \hat{\omega}_0 < \omega_s\}$  is small, which it is when the SNR is large enough, the  $E\{\hat{\omega}\} \approx \omega_0$  or  $\hat{\omega}_0$  is unbiased. If  $\Pr\{2\omega_0 < \hat{\omega}_0 < \omega_s\}$  is significant then  $\hat{\omega}_0$  is biased in the direction of  $\omega_s/2$ . In other words,  $E\{\hat{\omega}_0 - \omega_0\} > 0$ . If  $\omega_0 > \omega_s/2$  the above remarks reply with the obvious modifications. Observe that due to the symmetry of the problem, the bias of  $\hat{\omega}_0$  must be an odd function of  $\omega_0$  about  $\omega_s/2$ .

It is easy to show that if  $\omega_0$  is equal to zero,  $\omega_s/2$ , or  $\omega_s$  the p.d.f. of  $\hat{\omega}_0$  is even about  $\omega_s/2$ . Thus, in these three cases  $E(\hat{\omega}_0) = \omega_s/2$ . We see, therefore, that the bias of  $\hat{\omega}_0$  has the following values:

$\omega_0$	$E(\hat{\omega}_0 - \omega_0)$
0	$\omega_s/2$
$\omega_s/2$	0
$\omega_s$	$-\omega_s/2$

Clearly we expect to make frequency estimation errors if  $\omega_0$  is close

to zero or  $\omega_s$ . At moderate SNR, say above the threshold region, we would not expect trouble if the difference between  $\omega_0$  and zero (or  $\omega_s$ ) is at least four times the RMS bound. In practice we do not expect the ambiguity at zero at  $\omega_s$  to be a problem.

The variance of  $\hat{\beta}$  depends only upon the SNR. Thus the variance of  $\hat{\omega}_0$  is proportional to  $\omega_s^2$  and is not a function of  $\theta_0$ . The variance of  $\hat{\omega}_0$  is a function of  $\omega_0$ , but its variation with  $\omega_0$  is small at SNR above the threshold region. Hence the C-R bound for unbiased estimators is appropriate in this region.

An example. When there is no noise then it can be shown that

$$A(\omega) = b_0 e^{j\theta_0} e^{-j(N-1)z} \frac{\sin Nz}{N \sin z}, \quad (45)$$

where

$$z = \frac{\omega - \omega_0}{2} T = \pi (\omega - \omega_0) / \omega_s. \quad (46)$$

$B(\omega)$  is  $|A(\omega)|^2$ . Thus

$$B(\omega) = b_0^2 \frac{\sin^2(Nz)}{N^2 \sin^2(z)}. \quad (47)$$

This function is shown on Figure 3. The function is symmetric about  $\omega_0$  and has period  $\omega_s$ . The global maximum occurs at  $\omega_0$  and has value  $b_0^2$ . Notice numerous low-amplitude maxima. Without noise the ML estimates of  $\omega_0$  and  $b_0$  have no error.

When noise is present  $B(\omega)$  loses its clean, symmetrical shape and the minor maxima get larger. The global maximum is usually close to  $\omega_0$ . Figure 4 illustrates this situation.

If the SNR is small  $B(\omega)$  will occasionally be so badly distorted that

the global maximum occurs at a frequency far removed from  $\omega_0$  as illustrated in Figure 5. When this happens, the ML frequency estimation algorithm makes a large error. It is the occurrence of these rare but large errors, which we call sports, at low SNR that causes  $\text{VAR}\{\hat{\omega}_0\}$  to pull away from the C-R bound.

### AN ALGORITHM

As indicated above, once an estimate of  $\omega_0$  is made, estimates of  $b_0$  and  $\theta_0$  can be done by straight-forward computations, using appropriate equations. Thus the difficult and time-consuming part of an algorithm is the part that locates the maximum of  $B(\omega)$ . This part is essentially a search routine.

One way to develop an algorithm is to use a two-part search routine. The first part calculates  $B(\omega)$  for a set of  $\omega$  values between zero and  $\omega_s$ , and identifies the  $\omega$  that maximizes  $B(\omega)$  over this set of  $\omega$  values. The second part locates the local maximum closest to the value of  $\omega$  picked out by the first part. We call the first part the coarse search and the second part the fine search.

#### The Coarse Search

For the coarse search we evaluate  $B(\omega)$  at the set of frequencies  $\{\omega_k\}$  defined by

$$\omega_k = \frac{2\pi k}{MT}, \quad k = 0, 1, 2, \dots, M-1, \quad (48)$$

where  $M$  is a power of 2 greater than or equal to the number of samples,  $N$ . We also always choose  $N$  to be a power of 2. Thus  $M/N$  is also a power of 2.

Observe that the set  $\{A(\omega_k)\}$  is the DFT of the set  $\{\bar{Z}_n\}$  defined by

$$\bar{Z}_n = \begin{cases} Z_n, & n = 0, 1, 2, \dots, N-1 \\ 0, & n = N, N+1, \dots, M-1 \end{cases} \quad (49)$$

The set  $\{B(\omega_k)\}$  is simply the set  $\{|A(\omega_k)|^2\}$

The output of the coarse search is the value of  $\omega_k$ , say  $\omega_\ell$ , that corresponds to the largest member of the set  $\{B(\omega_k)\}$ .

It is natural to use  $M = N$  in the coarse search. However, it turned out that the  $\omega_\ell$  thus obtained was the wrong choice (not close to the global maximum) often enough at low SNR to cause trouble. We found that the number of wrong choices was significantly reduced when the coarse search used  $M/N$  equal to 2 or 4.

Our coarse search uses a fast Fourier transform (FFT) algorithm to compute the desired DFT.

### The Fine Search

The fine search algorithm locates the value of  $\omega$  closest to  $\omega_\ell$  that maximizes  $B(\omega)$ . If the derivative of  $B(\omega)$  at  $\omega_\ell$ ,  $B'(\omega_\ell)$  is positive, the desired maximum is at a value of  $\omega$  greater than  $\omega_\ell$ . Otherwise the desired maximum is at a value of  $\omega$  less than or equal to  $\omega_\ell$ .

Consider Figure 6. Given a frequency to start from  $\omega_\ell$ , the problem is to locate the closest zero in  $B'(\omega)$  with  $B''(\omega) < 0$ . In the figure this occurs at point B. Points A and C correspond to minima in  $B(\omega)$ .

Our fine search algorithm computes  $B'(\omega)$  at points  $P_1$  through  $P_8$ , finally locating points on either side of the desired zero, points  $P_7$  and  $P_8$ .

Then we use the secant method (Action [12], p. 52) to compute successive approximations to point B and the frequency estimate,  $\hat{\omega}$ . The iteration formula is simple:

$$\omega_{i+1} = \frac{B'(\omega_{i-1})\omega_i - B'(\omega_i)\omega_{i-1}}{B'(\omega_{i-1}) - B'(\omega_i)} \quad (50)$$

The process is stopped if  $B'(\omega_i) = B'(\omega_{i-1})$ , if  $B'(\omega_i) = 0$ , or if  $|\omega_{i+1} - \omega_i|$  is less than 1 percent of the square root of the bound on the variance of  $\hat{\omega}_i$ .

If  $\omega_\ell < \hat{\omega}$  then the initial steps of the fine search work to the left of  $\omega_\ell$ , using  $B'(\omega_\ell) < 0$  to indicate that  $\hat{\omega} < \omega_\ell$ . In either case the initial steps are small enough to avoid missing the correct zero in  $B'(\omega)$ ; the step size is  $\frac{\omega_s}{5N}$ .

### Threshold Effect

It is well known that nonlinear estimation is generally plagued by threshold effects. At low SNR there is usually a range of SNR in which the mean squared error (m. s. e.) rises very rapidly as SNR decreases. The SNR at which this effect is first apparent is called the threshold. Receivers are often said to operate above or below threshold.

Digital frequency estimation also has threshold effects, generally connected with the occurrence of sports. In this section we present a calculation of threshold effects. The result accurately describes one particular model.

Consider the estimation of the frequency of a single complex tone. Let the sampling frequency be  $\omega_s$  and the number of samples be  $N$ . Assume the phase is unknown. Suppose the tone frequency is  $\omega_0 = \frac{\omega_s}{2}$ . Assume the algorithm is the following, using  $M = N$ :

1. compute the set:  $A_k = \frac{1}{N} \sum_{n=0}^{N-1} Z_n e^{-j2\pi nk/N}$ . This is the DFT of  $\{Z_n\}$ .
2. Identify the largest  $|A_k|$ , say  $|A_\ell|$ , and record  $\ell$ .
3. Start the fine search at  $\omega = \omega_\ell$  and continue until the maximum of  $B(\omega)$  is found.

Since we chose  $\omega_0 = \frac{\omega_s}{2}$ ,  $A_{N/2}$  should be the largest. That is, the coarse search should give  $\ell = N/2$ . If  $\ell \neq N/2$  we say a sport has occurred. If  $\ell \neq N/2$  the m. s. e. is greater than zero and less than the square of the distance to  $\omega_{\ell+1}$ .

We will approximate the m. s. e. in this case by the C-R bound for an unbiased estimator, which we designate  $\omega_{CR}^2$ . From equation (23),

$$\omega_{CR}^2 = \frac{3\omega_s^2}{2\pi^2 \rho N(N^2 - 1)} \quad (52)$$

where

$$\rho = b_{0/2\sigma}^2. \quad (53)$$

If a sport occurs, the outcome of the fine search will be any frequency between zero and  $\omega_s$ . The p. d. f. is approximately uniform because the signal has little influence. Thus we write the m. s. e. when a sport occurs as

$$\omega_{sp}^2 = \frac{\omega_s^2}{12} \quad (54)$$

In the general case where the signal frequency is not equal to  $\omega_s/2$ , equation (54) would become



$$\omega_{sp}^2 = \frac{\omega_s^2}{12} + \left[ \omega_1 - \frac{\omega_s}{2} \right]^2 \quad (55)$$

The total m. s. e. is the weighted sum of the two contributions,

$$\text{m. s. e.} = (\text{m. s. e. /sport}) p + (\text{m. s. e. /no sport})(1-p) \quad (56)$$

Let the total m. s. e. be  $\omega_e^2$ . Then we have

$$\omega_e^2 \approx p \frac{\omega_s^2}{12} + (1-p) \frac{3\omega_s^2}{2\rho\pi^2 N(N^2-1)} \quad (57)$$

The RMS error is

$$\omega_{\text{RMS}} = \sqrt{\omega_e^2} \quad (58)$$

Next we calculate the probability of a sport,  $p$ , and verify that when a sport occurs all possible  $\ell$  except the correct one are equally likely.

Probability of a Sport. Let  $C_k = |A_k|$ ,  $k = 0$  to  $N - 1$ , (59)

where  $A_k$  was defined above. When both signal and noise are present, each  $C_k$  is a random variable. If the signal frequency is  $\frac{\omega_s}{2}$  and the noise samples are independent, normal, and zero-mean with variance  $\sigma^2$  then it can be shown that the  $C_k$  are independent with Rayleigh distribution: [10]

$$f_k(C_k) = \frac{NC_k}{\sigma^2} e^{-\frac{NC_k^2}{2\sigma^2}}, \quad C_k \geq 0, \quad k \neq N/2 \quad (60)$$

Let  $r = N/2$ .  $C_r$  has a Ricean distribution: [10]

$$f_r(C_r) = \frac{NC_r}{\sigma^2} e^{-\frac{N(C_r^2 + b^2)}{2\sigma^2}} I_0 \left[ \frac{NbC_r}{\sigma^2} \right], \quad C_r \geq 0, \quad (61)$$

where  $b$  is the signal amplitude. Then

$$\begin{aligned} 1 - p &= P_r \{ \text{all } C_k < C_r \} \\ &= \int_x P_r \{ \text{all } C_k < C_r / C_r = x \} P_r \{ C_r = x \} dx. \end{aligned} \quad (62)$$

But  $P_r \{ \text{all } C_k < C_r / C_r = x \} = [P_r \{ C_1 < C_r / C_r = x \}]^{N-1}$

Thus

$$1 - p = \int_0^x f_r(x) \left[ \int_0^x f_k(y) dy \right]^{N-1} dx. \quad (63)$$

$$\begin{aligned} \int_0^x f_k(y) dy &= \int_0^x \frac{Ny}{\sigma^2} e^{-Ny^2/2\sigma^2} dy \\ &= 1 - e^{-Nx^2/2\sigma^2}. \end{aligned} \quad (64)$$

Then

$$1 - p = \int_0^\infty \left[ 1 - e^{-\frac{NX^2}{2\sigma^2}} \right]^{N-1} \frac{NX}{\sigma^2} e^{-\frac{N(X^2+b^2)}{2\sigma^2}} I_0 \left[ \frac{NbX}{\sigma^2} \right] dx. \quad (65)$$

After some further work, we obtain

$$1 - p = \sum_{k=0}^{N-1} \frac{(N-1)! (-1)^k}{(N-1-k)! k! (k+1)} e^{-N\rho k/k+1} \quad (66)$$

and

$$p = \frac{1}{N} \sum_{m=2}^N \frac{N! (-1)^m}{((N-m)! m!)} e^{-N\rho \frac{m-1}{m}} \quad (67)$$

It is easy to verify that the limit of  $p$  as  $\rho \rightarrow 0$  is  $\frac{N-1}{N}$ , which is in agreement with our assumptions. The formulas for  $p$  given above are neat closed forms that cannot be summed on a computer because the terms

$\frac{N!(-1)^m}{(N-m)!m!}$  get very large and alternate in sign. Thus to compute  $p$  it was necessary to use the integral form and do numerical integration. When  $N\rho$  is large one can use the first term of  $p$ :

$$p \approx \frac{N-1}{2} e^{-N\rho/2}, \quad p < 10^{-3}. \quad (68)$$

The calculated values of  $p$  are shown on Figure 7.

Approximate RMS Frequency Error. We used the above formula for several values of  $N$  as shown on Figure 8. The small circles of the curves represent the results of simulations. As can be seen, the simulation results agree with the calculated curves. The curves are similar to the well known results for the continuous observation case. See Van Trees, p. 285.

On Figure 8 the simulation for  $N = 16$  each were done with 800 estimations. The ones for  $N = 128$  were done with 500 estimations. Five hundred estimations will have a sample variance within 12.5 percent of the true variance with the same confidence.

One would not usually operate a system at SNR below the threshold. Thus Figure 8 is useful mainly because it shows the SNR at which the threshold effect starts. All SNR above threshold can be considered to be "high SNR" in the sense that the variance of ML estimators equals the C-R bounds at high SNR.

#### Level Estimates

The simulations described above included level estimates according to equation (33). In every case the RMS level errors were almost equal to the C-R bounds. Threshold effects were not observed!

#### Remarks

We ran the above-described simulations with  $M = 4N$  instead of  $M = N$ .

There was no significant difference in the results. Since using  $M = 4N$  is more likely to result in correctly locating the global maximum in  $B(\omega)$ , we are led to believe that Figure 8 truly depicts ML estimation when the frequency is one half the sampling frequency.

The next question is, what about different signal frequencies? We ran the simulation with  $M = 64$ ,  $N = 16$ , and  $f_0 = 2120$  Hz, using -10, -5, 0, and 5 db SNR. The only point different from the 0 points is the  $\square$  point on Figure 8. As before, level estimates did not show a threshold effect.

### SUMMARY

This has been a quick trip through a study of the problem of estimating the frequency and level of a cycloidal signal from a finite number of noisy observations of the signal. We derived the equations that describe the Cramer-Rao lower bounds to the variance of estimation errors. Then we derived the maximum-likelihood estimators and showed their relationship to the discrete Fourier transform. The analysis of the ML estimators revealed some of their properties. Then we looked at an algorithm suitable for implementation on a digital computer. The algorithm almost always yields ML estimates. We were able to derive an expression for the threshold behavior of the algorithm. Simulation results verified the analysis.

The overall conclusion for the case studied is that ML estimates is feasible and will yield estimates which are as good as permitted by the C-R bounds (above threshold).

The general cases of real tones (sinusoidal signals) and of many tones are, in a sense, an extension of the case studied here. The presence of several cycloidal signals introduces complexity in the bounds, ML estimation, and practical algorithms. These matters have been studied [10] but are not reported here.

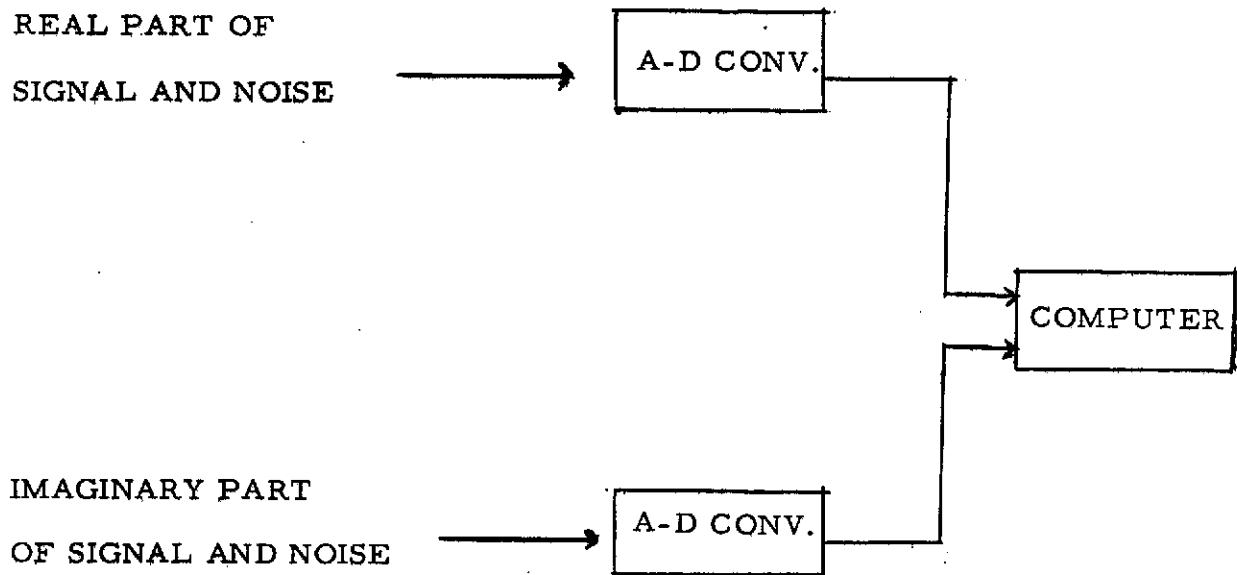


FIGURE 1 - PARAMETER ESTIMATION SYSTEM MODEL

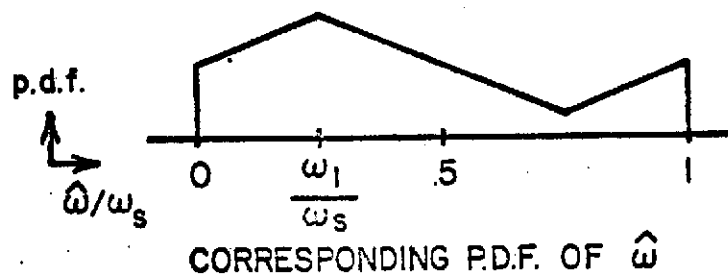
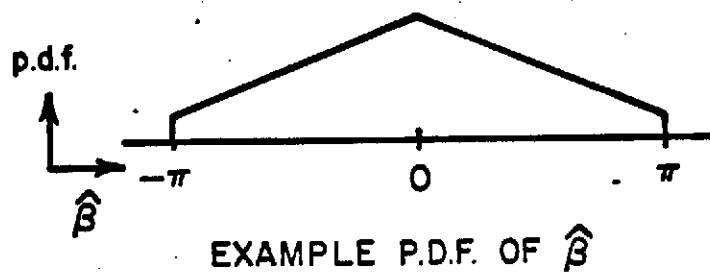


FIGURE 2 - Relationship of p. d. f. of  $\hat{\beta}$  to p. d. f. of  $\hat{\omega}$ .

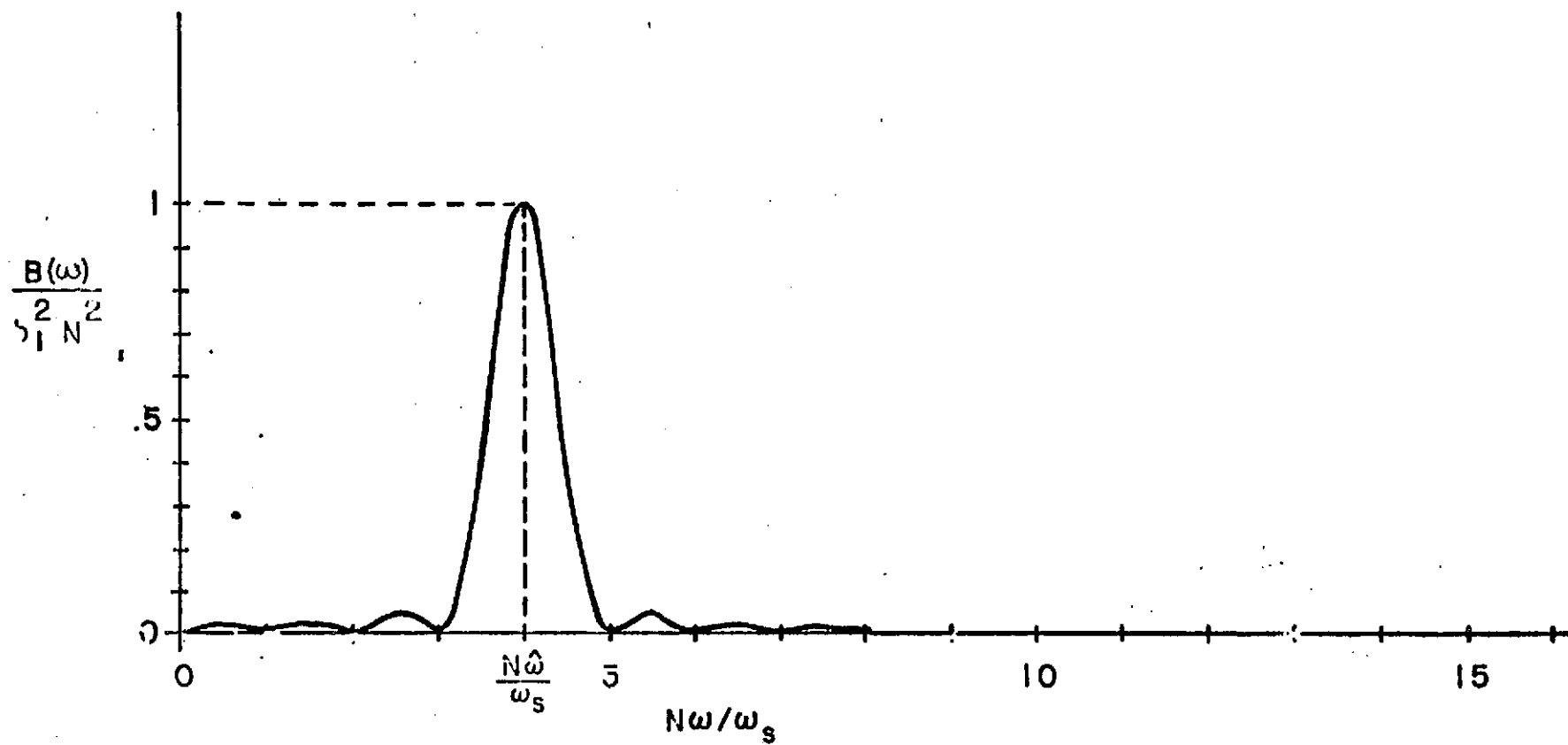


FIGURE 3 - Shape of  $B(\omega)$  from single complex tone without noise.  $N$  is 16.

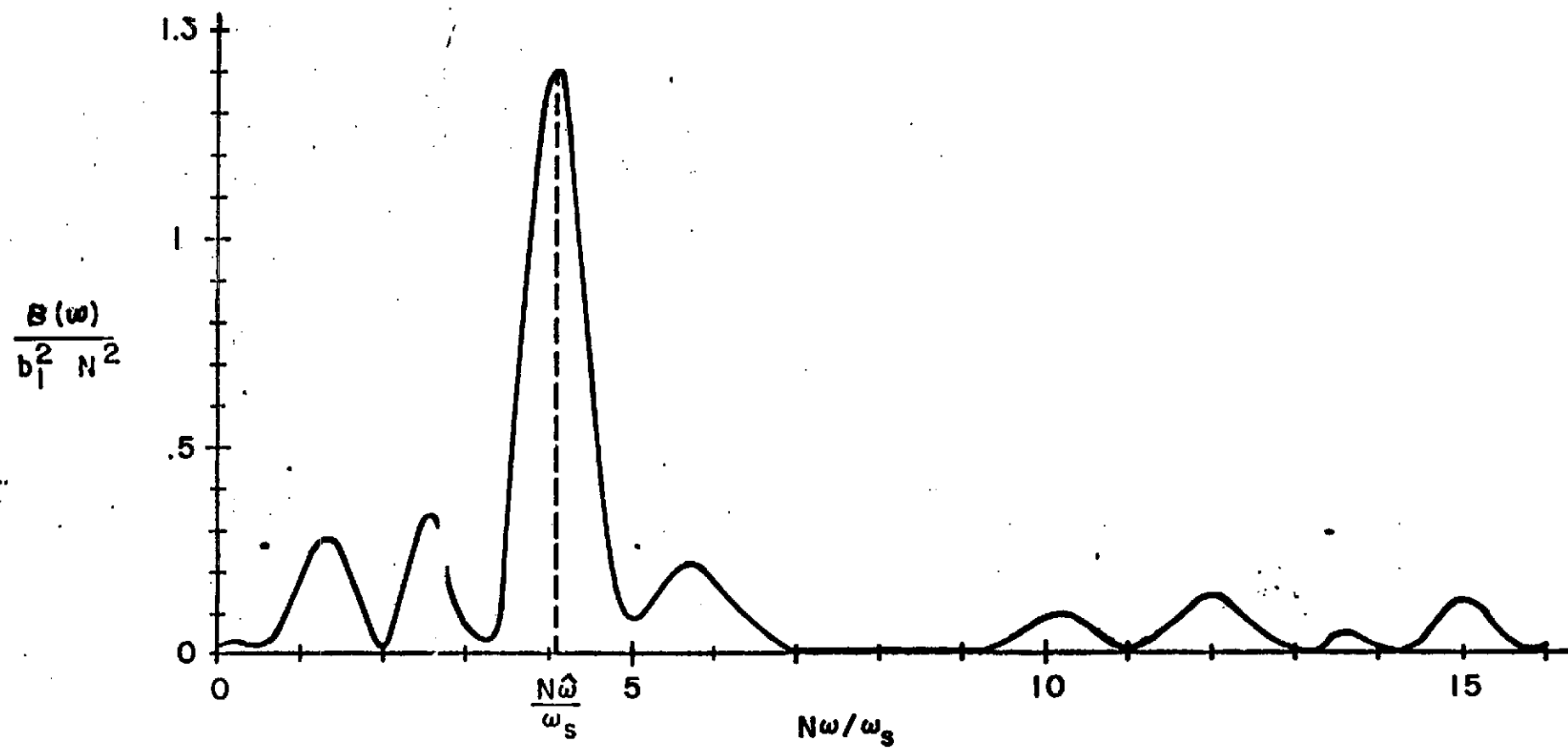


FIGURE 4 - An example of  $B(\omega)$  for a single complex tone at zero-db SNR.  $N$  is 16 .



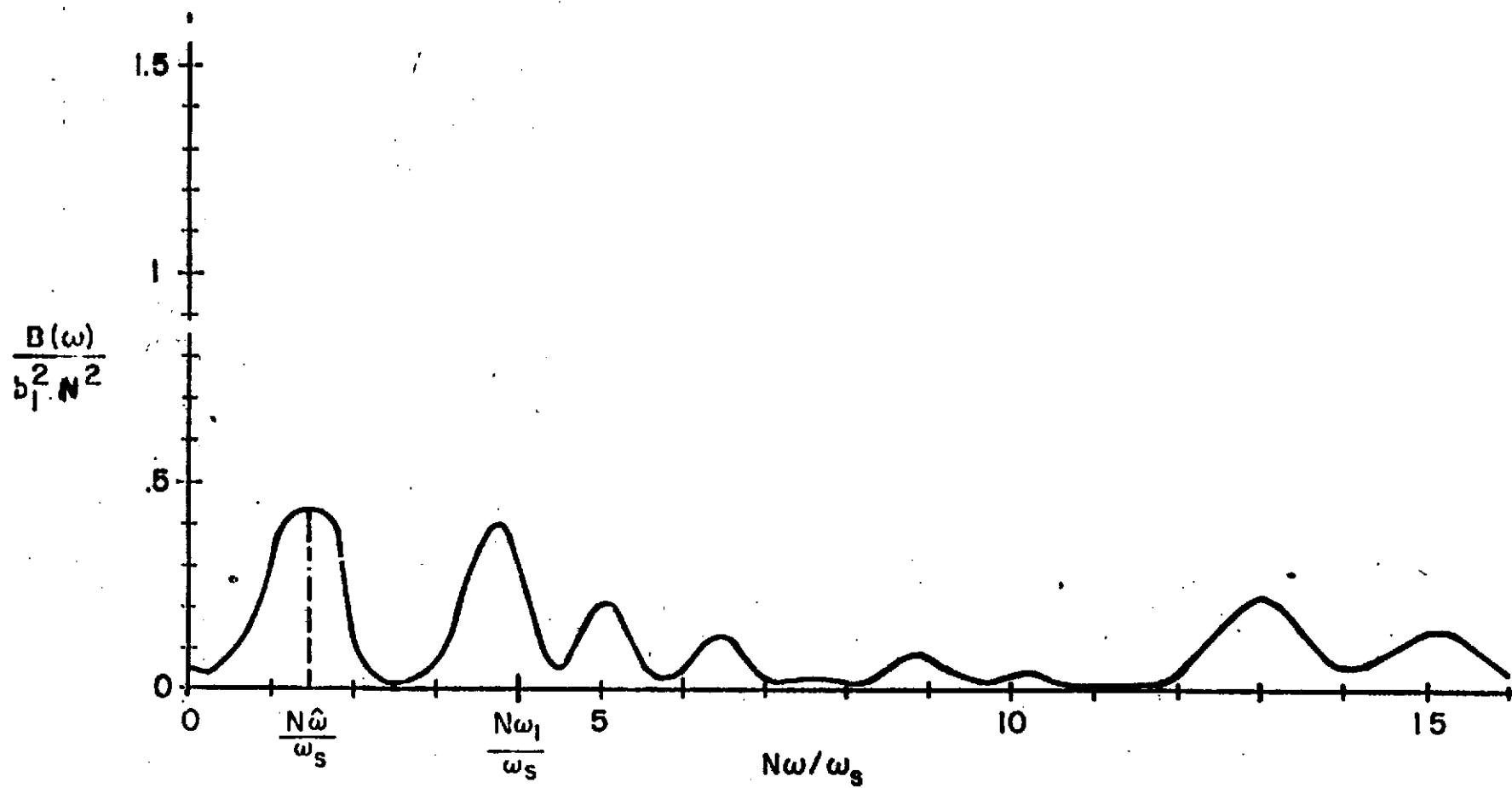


FIGURE 5 - An example of  $B(\omega)$  for a single complex tone at zero-db SNR, showing a sport.  $N$  is 16.

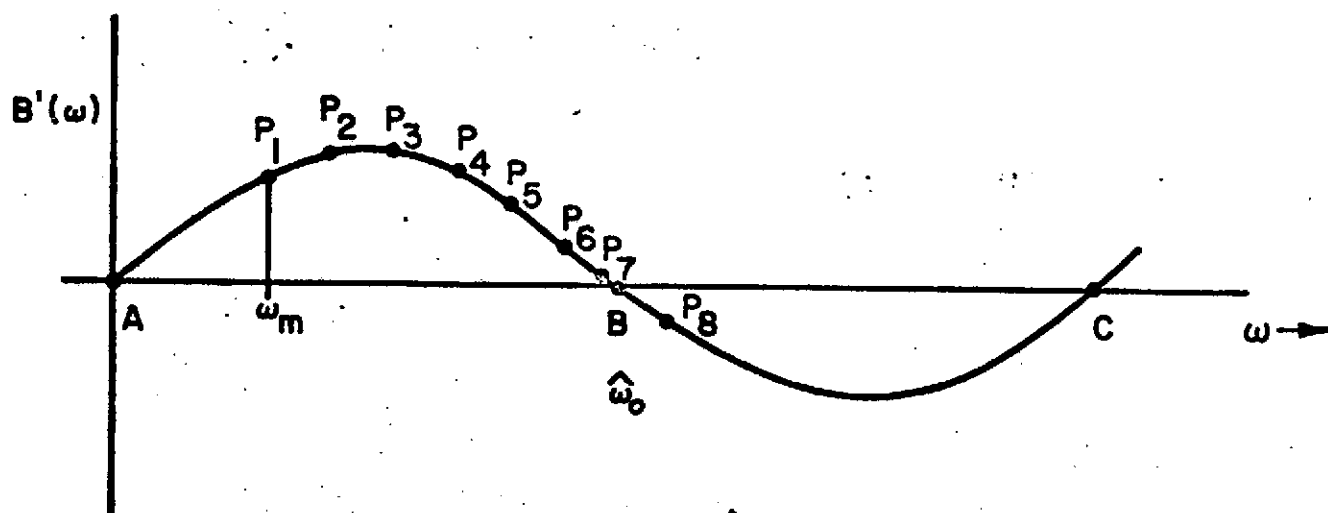


FIGURE 6 - Illustration of search for zero of  $B'(\omega)$  .

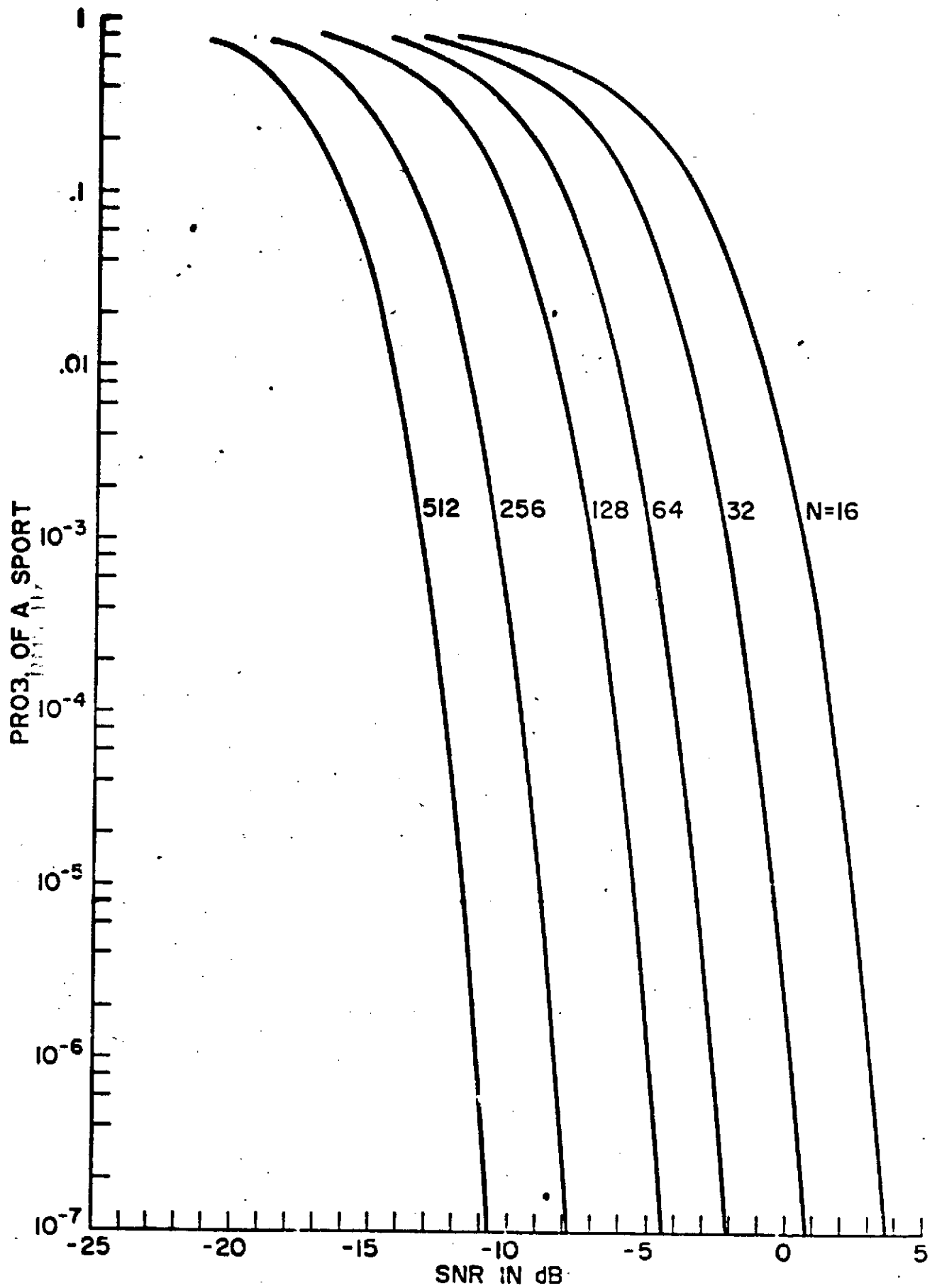


FIGURE 7 - Probability of a sport versus SNR.

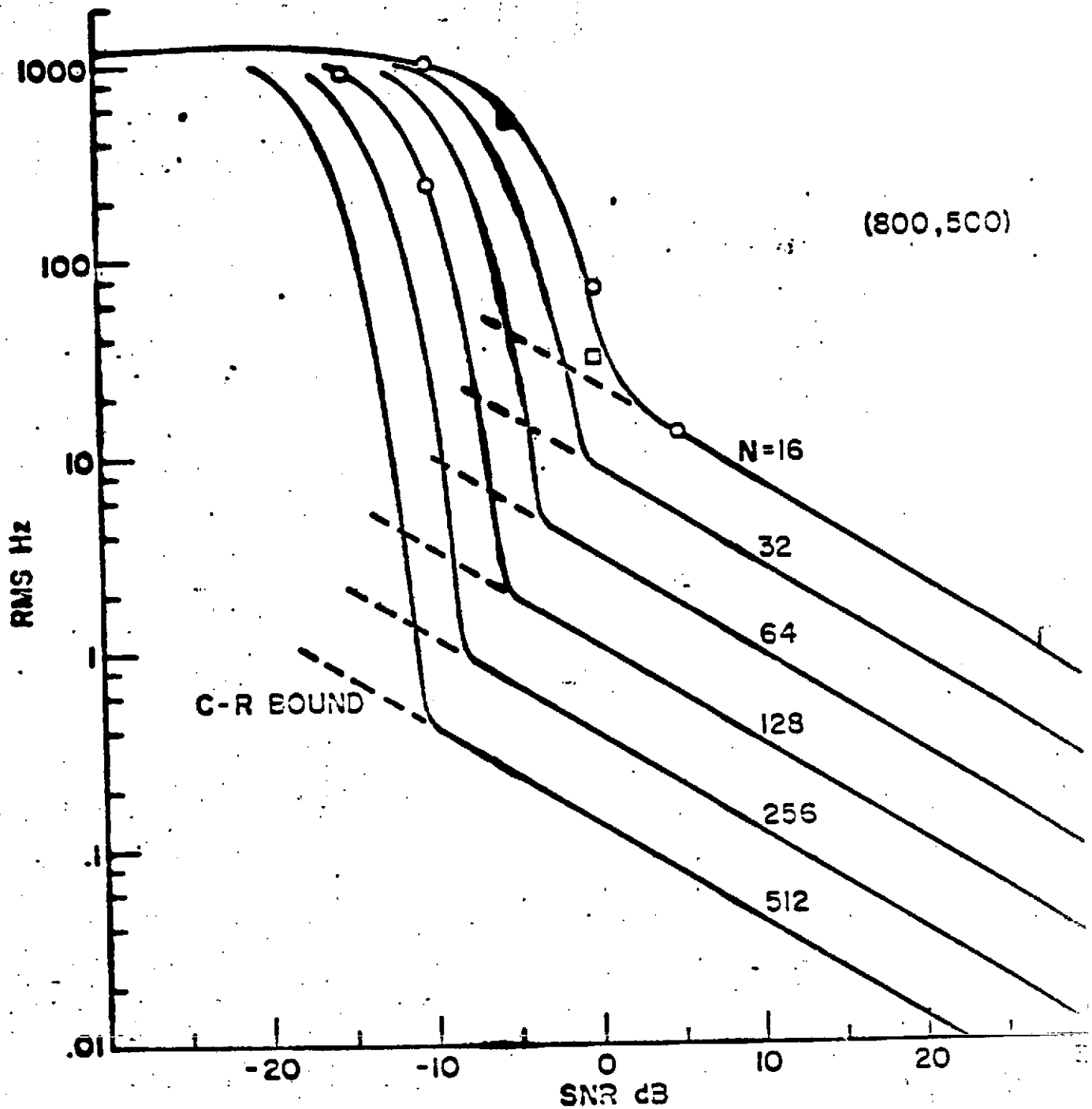


FIGURE 8 - Approximate performance of ML frequency estimate of single complex tone at 2000 Hz, with unknown phase.

# REFERENCES

1. Slepian, D., Estimation of Signal Parameters in the Presence of Noise, Trans. IRE Prof. Group on IT, PG IT-3, Vol. 68, pp. 68-89, March, 1954 .
2. Cramer, Harold, Mathematical Methods of Statistics, Princeton Univ. Press, 1946.
3. Rao, C.R., Minimum Variance and the Estimation of Several Parameters, Proc. Comb. Phil. Soc., 43(20), pp. 280-283, 1946 .
4. Bhattacharyya, A., On Some Analogues of the Amount of Information and Their Use in Statistical Estimation, Sankhya The Indian Journal of Statistics, Vol. 8, in four chapters, pp. 1-14, (1946), pp. 201-218 (1947) , and pp. 315-328 (1948) .
5. Barankin, E.W., Locally Best Unbiased Estimators, Ann. Math. Stat., Vol. 20, pp. 477-501, 1949 .
6. Glave, F.E., A New Look at the Barankin Lower Bound, IEEE Trans IT, IT-18, No 3, pp. 349-356, May, 1972 .
7. Ziv, J. and M. Zakai, Some Lower Bounds on Signal Parameter Estimation, IEEE Trans IT, Vol. 15, No. 3, pp. 386-391, May, 1969.
8. Chapman, D.G. and H. Robbins, Minimum Variance Estimation Without Regularity Conditions, The Annals of Math. Stat., Vol. 22, No. 4, pp. 581-586, December, 1951 .
9. Seidman, L.P. Performance Limitations and Error Calculations for Parameter Estimation, Proc. IEEE, Vol. 58, No. 5, pp. 644-652. May, 1970 .
10. Rife D.C., Digital Tone Parameter Estimation in the Presence of Gaussian Noise, PhD Thesis, Polytechnic Institute of Brooklyn, June, 1973.
11. Van Trees, Harry L., Detection, Estimation, and Modulation Theory, Part I., John Wiley and Sons, Inc., New York, 1968.
12. Cochran, W.T., et. al, What is the Fast Fourier Transform ?, IEEE Trans AE, AV-15, No. 2, pp. 45-55, June, 1967.
13. Cooley, J.W., P.A.W. Lewis, and P.D. Welch, The Fast Fourier Transform and its Applications, IBM Research Paper RC 1743, February 9, 1967.
14. Cooley, J.W., P.A.W. Lewis, and P.D. Welch, The Finite Fourier Transform, IEEE Trans AE, AV-17, No. 2, pp. 77-85, June, 1969.
15. Acton, F.S., Numerical Methods That Work, Harper and Row, New York, 1970.

### III. DIGITAL FILTERING FOR RADAR SIGNAL PROCESSING APPLICATIONS

A novel approach in synthesizing digital filters for signal processing applications is presented. This approach is an extension of the frequency sampling method of nonrecursive filter synthesis. Appropriate time delays are used in conjunction with a set of parallel complex exponential resonators whose outputs are summed to yield a desired filter impulse response. This synthesis method takes advantage of the known signal waveform structure and results in many fewer digital computations as compared to convolution processing. This approach is particularly suited to synthesis of matched filters for radar signal processing and yields matched or approximately matched filters which simultaneously have very low storage and very low computational requirements.

#### I. Introduction

Interest in the application of digital filtering to signal processing is increasing and numerous publications (Refs. 1-5) have been written on various approaches to digital signal processing. In this paper a novel approach to signal processing digital filter synthesis is presented which takes advantage of the signal waveform structure and results in a reduction in the required digital computations.

The signal to be extracted from an interference background often consists of only a few data points as compared to the amount of data to be processed. An example of this is found in radar signal processing where the data to be processed usually consists of thousands of samples, whereas the signal is composed of several hundred or fewer samples.

The input data to a radar analog signal processor consists of analog

voltage variations representing signal plus noise (or noise alone) corresponding to a given time or range extent. To use this information in a digital processor, these voltage variations should be sampled such that there is at least one sample per range resolution cell (defined as the reciprocal of the signal bandwidth). These samples are then converted into binary words which represent the amplitude and phase, or in-phase and quadrature components of the sampled analog voltages. These complex valued words or samples are then operated upon to extract the wanted signal from the background noise.

The radar signal to be extracted from noise generally has a time extent less than that of the input data of interest, but longer than a range resolution cell if the signal bandwidth-time (BT) product is greater than one. The presence and location of the signal in the input data can be determined in an optimum manner by convolving the complex conjugate of the sampled signal reversed in time,  $s^*(-n)$ , with the input data,  $y(n)$ , as follows<sup>†</sup>:

$$a(n) = \sum_{j=0}^n \chi(j) s^* [N-n+j] ; \quad n = 0, 1, 2, \dots, J-1 \quad (1)$$

where  $a(n)$  is the sampled cross-correlation function of  $\chi(n)$  and  $s^*(-n)$ . It is assumed that there are  $J$  range data samples and  $N$  signal samples. If the signal is present at some point in the input data, the cross-correlation function<sup>††</sup> will peak up at that range location.

---

<sup>†</sup>It can be shown that, for purposes of signal detection and signal parameter estimation, the signal can be extracted from Gaussian-distributed interference in an optimum manner by matched or correlation filtering (Ref. 6). This type of filtering involves the convolution of the signal and data as described above.

<sup>††</sup>Since the  $\chi(j)$ ;  $j=0, 1, \dots, n$  in (1) are random variables, then each  $a(n)$  is also a random variable. Therefore, the discrete function  $\{a(n); n=0, 1, \dots, J-1\}$  is a sample from an ensemble of discrete random functions.

This type of signal processing is referred to as "Moving Window Convolution Processing" since the finite-length samples signal reversed in time,  $s^*(-n)$ , acts as a window moving past the range data. Note that each value of  $a(n)$  is formed by one position of the "moving window" relative to the range data. If  $J$  samples of range data are present, there must be  $J$  different positions of the "moving window" in order to produce  $J$  different values of  $a(n)$ .

Another method of extracting the signal from the range data is to perform the filtering in the frequency domain. That is, transform all of the range (time) data to the frequency domain by use of the Digital Fourier Transform (DFT) as follows:

$$X(k) = \sum_{n=0}^{J-1} x(n) \exp \left[ -j \frac{2\pi kn}{J} \right], \quad k = 0, 1, 2, \dots, J-1 \quad (2)$$

Then, perform the filtering by multiplying the frequency data by the Fourier Transform of  $s^*(-n)$ :

$$Y(k) = S^*(k) X(k); \quad k = 0, 1, 2, \dots, J-1 \quad (3)$$

The resulting frequency samples are then transformed back into the time domain by use of the Inverse Digital Fourier Transform (IDFT):

$$a(n) = \frac{1}{J} \sum_{k=0}^{J-1} Y(k) \exp \left[ j \frac{2\pi kn}{J} \right]; \quad n = 0, 1, 2, \dots, J-1 \quad (4)$$

If the signal is present, the cross-correlation function<sup>††</sup>,  $a(n)$ , will peak up at that range location.

If this method of processing were used, the Fast Fourier Transform (FFT) and Inverse Fast Fourier Transform (IFFT) would probably be used instead of the DFT and IDFT as indicated above. The transformations accomplished by the DFT and FFT, or the IDFT and IFFT, are identical;

---

†† See note on previous page.



the way in which the transformations are accomplished is different (Ref. 7).

This type of processing is referred to as "batch processing", since the complete set or "batch" of range data is processed at once rather than in subsets as in moving window convolution processing.

Batch processing is sometimes more desirable than moving window convolution processing because many fewer digital computations are required when the Fast Fourier Transform (FFT) is used. However, the disadvantage of batch processing is that a large amount of storage is required to store the thousands of range data points. To date, batch processing for many radar applications is too expensive to implement because of this large storage requirement.

On the other hand, moving window convolution processing requires less storage (only the number of signal samples) but more digital computations and is, therefore, also very expensive to implement. However, if the number of required computations in moving window processing could be reduced significantly, then the combination of relatively low storage and low computational requirements would make it a desirable approach to digital signal processing. Hence, it is of interest to consider ways of reducing the number of digital computations required in moving window processing.

In the following sections of this paper, a synthesis technique is discussed which takes advantage of the signal waveform structure such that the number of digital computations required is at least an order of magnitude lower than that required in moving window convolution processing. In addition, the storage required with this technique is about the same as in moving window convolution processing.

In order to explain this technique clearly, several methods of synthesizing digital filters are briefly reviewed in the next section of this paper.

## II. Methods of Nonrecursive Filter Synthesis

In the digital filter synthesis problem, the desired impulse response is assumed to be known and is represented by the sequence

$$\{ h(n): n = 0, 1, 2, \dots \} \quad (5)$$

where each  $h(\cdot)$  is complex valued. \* This discussion will be restricted to the synthesis of nonrecursive digital filters which are characterized by a finite-duration impulse response such as is required in moving window processing. Hence, the impulse response will be represented by a sequence of  $N$  complex numbers

$$\{ h(n): n = 0, 1, 2, \dots, N-1 \} \quad (6)$$

A straightforward method of synthesizing a filter with this type of impulse response is the convolution or "tapped delay-line" filter realization illustrated in Figure 1. In this realization the  $N$  weights are the complex values of the  $N$  samples in the filter's impulse response, and each of the  $N-1$  delays used corresponds to the delay between the filter impulse response samples.

Now, if the impulse response is required to be

$$h(n) = s^{*(N-n)}; \quad n = 0, 1, 2, \dots, N-1 \quad (7)$$

then the output of the convolution filter of Figure 1 is related to the input data,  $x(n)$ , as given by Equation (1). Hence, this filter would perform the moving window processing desired.

---

\* In radar/sonar signal processing the number sequences encountered are complex valued, hence, the need to assume  $h(\cdot)$  is complex.

The number of complex multiplications required with this filter is  $N \cdot K$  where  $N$  is the number of samples in the signal (and in the filter impulse response),  $K$  is the number of data samples to be processed, and where the number of data samples is assumed to be greater than the number of signal samples ( $K > N$ ).

Now let us consider another method of synthesizing a digital filter with the impulse response of Equation (6).

Since the impulse response is of finite duration, it can be represented in terms of its Discrete Fourier Transform (DFT)  $\{H_k : k = 0, 1, 2, \dots, N-1\}$  as follows:

$$h(n) = \frac{1}{N} \sum_{k=0}^{N-1} H_k \exp \left[ \frac{j2\pi kn}{N} \right]; \quad n = 0, 1, 2, \dots, N-1 \quad (8)$$

The  $z$ -transform of the sequence of (6) is

$$H(z) = \sum_{n=0}^{N-1} h(n) z^{-n} \quad (9)$$

Substituting (8) into (9) and interchanging sums, the sum over the  $n$  index can be evaluated in closed form so that

$$H(z) = \sum_{k=0}^{N-1} \frac{H_k}{N} \frac{1 - z^{-N}}{1 - z^{-1} \exp \left[ \frac{j2\pi k}{N} \right]} \quad (10)$$

Evaluating (10) on the unit circle where  $z = \exp[j2\pi f\tau]$  leads to the frequency response

$$H(f) = \exp[-j(N-1)\pi f\tau] \cdot \sum_{k=0}^{N-1} \frac{H_k}{N} \cdot \exp \left[ -\frac{j\pi k}{N} \right] \cdot \frac{\sin N\pi f}{\sin \pi \left[ f\tau - \frac{k}{N} \right]} \quad (11)$$

The symbol  $\tau$  denotes the time spacing of the number sequence in (6) and is often assumed to be unity. Note that when  $z = \exp \left[ \frac{j2\pi k}{N} \right]$ ,  $H(z) = H_k$

or, equivalently, when  $f = \frac{k}{N\tau}$ ,  $H(f) = H_k$ .

The interpretation of (10) and (11) is as follows: the frequency response of the digital filter which is described by the complex impulse response in (6) is a sum of  $\frac{\sin Nx}{\sin x}$  frequency responses, each with a complex weight  $H_k$  and a center frequency  $\frac{k}{N\tau}$  where  $k = 0, 1, 2, \dots, N-1$ , as illustrated in Figure 2. A realization of the filter of Equation (10) is shown in Figure 3. Each block in this figure represents a zonal filter with the appropriate center frequency and bandwidth. Since the term  $(1-z^{-N})$  is common to all of the filter functions, it can precede the filter bank and be shared.

The parameters needed to completely specify the digital filter are either the sequence  $\{H_k; k = 0, 1, 2, \dots, N-1\}$  and the center frequencies  $\{f_k = \frac{k}{N\tau}; k = 0, 1, 2, \dots, N-1\}$  or the sequence  $\{h(n); n = 0, 1, 2, \dots, N-1\}$  and the time delay between samples,  $\tau$ . If the desired frequency response is known, the filter weights are determined by sampling the desired frequency response at the frequencies  $f_k$ ; the complex values of these frequency samples are then used as the filter weights,  $H_k$ . On the other hand, if the desired impulse response is known, the filter weights,  $H_k$ , can be determined by use of the following equation, which is obtained from (8):

$$H_k = \sum_{n=0}^{N-1} H(n) \exp \left[ \frac{-j2\pi kn}{N} \right]; \quad k = 0, 1, 2, \dots, N-1 \quad (12)$$

Note that the impulse response of the filter of Figure 3 is exactly that given in Equation (6) if the complex weights are selected correctly. In other words, the samples taken from the frequency spectrum of  $h(n)$  are adequate to completely describe the filter frequency spectrum.

It will be useful for later discussions to consider in more detail the impulse response of the realization given in Figure 3. The impulse response of the  $l^{\text{th}}$  block or subfilter of this filter is

$$\exp\left[\frac{j2\pi\ell n}{N}\right]; n = 0, 1, 2, \dots; \ell = 0, 1, 2, \dots, N-1 \quad (13)$$

which is of infinite-time duration. However, when the term  $(1-z^{-N})$  is included with the  $\ell^{\text{th}}$  subfilter such that the z-transform of that filter is

$$\frac{1 - z^{-N}}{1 - z^{-1} \exp\left[\frac{j2\pi\ell}{N}\right]} \quad (14)$$

the corresponding impulse response is

$$\exp\left[\frac{j2\pi\ell n}{N}\right]; n = 0, 1, 2, \dots, N-1; \ell = 0, 1, 2, \dots, N-1 \quad (15)$$

which is of finite-time duration. Therefore, the impulse response of each subfilter or block of Figure 3 can be thought of as a time response with a rectangular amplitude and linear phase function that is sampled at the time instants  $nT$ ,  $n = 0, 1, 2, \dots, N-1$ . The overall impulse response of the digital filter is the weighted sum of the  $N$  time sequences described by (15), where  $\ell$  runs from 0 to  $N-1$ . The composition of the composite filter response is illustrated in Figure 4, where  $N$  is taken as four.

In some applications, one would expect some of the complex weights  $\{H_k; k = 0, 1, 2, \dots, N-1\}$  to be zero. For example, if the desired impulse response  $\{h(n); n = 0, 1, 2, \dots, N-1\}$  represented data with a small bandwidth (the complex value of the sequence varies slowly with increasing  $n$ ), many of the complex weights would be zero. A very special case would be where the magnitude of the complex impulse response did not vary with increasing  $n$  and the phase of the impulse response increased with  $n$  in a linear fashion; that is, the desired impulse response is given by

$$H_k \cdot \exp\left[\frac{j2\pi kn}{N}\right]; n = 0, 1, 2, \dots, N-1 \quad (16)$$

where the index  $k$  may take on any one of the discrete values  $0, 1, 2, \dots, N-1$ .

The resulting filter realization would have only one block or subfilter, as indicated in Figure 5.

The number of complex multiplications required in the filter realization of Figure 3 is  $2 \cdot N \cdot K$ , where  $N$  is the number of signal samples (and filter impulse response samples),  $K$  is the number of data samples to be processed, and  $K$  is assumed to be larger than  $N$ . The number of complex multiplications required for each subfilter (including the complex weights) is  $2 \cdot K$ . Therefore, assuming that all  $N$  of the subfilters are needed,  $N \cdot (2 \cdot K)$  complex multiplications are required in the composite filter realization.

In comparing the two filter realizations discussed thus far, one concludes that the use of the convolution filter realization of Figure 1 requires fewer multiplications than the frequency response realization and is, therefore, simpler and cheaper to implement. However, a further consideration of the frequency response filter realization leads one to a third filter realization that requires many fewer multiplications than even the convolution filter for some types of radar signals. This is the subject of the remainder of this paper.

### III. Synthesis of Specialized Digital Filter Response

In light of the discussion of the previous section, consider the digital filter illustrated in Figure 6. This filter is very much like the filter of Figure 3 except for the time delay following each subfilter. However, the impulse response of this filter is much different from the filter of the previous discussion.

The  $z$ -transform representing this filter function is

$$F(z) = \sum_{\ell=0}^{L-1} z^{-\ell N/L} \cdot \frac{F_{\ell}}{\frac{N}{L}} \left[ \frac{1 - z^{-\frac{N}{L}}}{1 - z^{-1} \exp \left[ \frac{j2\pi \ell L}{N} \right]} \right] \quad (17)$$

where  $N$  is the number of samples in the filter's impulse response,  $L$  is the number of subfilter, and  $\frac{N}{L}$  is assumed to be an integer. Evaluating (17) on the unit circle where  $z = \exp(j2\pi f\tau)$  leads to the frequency response

$$F(f) = \exp \left\{ -j\pi \left[ \left[ \frac{N}{L} \right] - 1 \right] f\tau \right\} \cdot \sum_{\ell=0}^{L-1} \frac{F_{\ell}}{\left[ \frac{N}{L} \right]} \cdot \exp \left\{ -j\pi \left[ 2\ell f\tau \left[ \frac{N}{L} \right] + \frac{\ell L}{N} \right] \right\}$$

$$\frac{\sin \left[ \frac{N}{L} \right] \pi f\tau}{\sin \pi \left[ f\tau - \frac{\ell L}{N} \right]} \quad (18)$$

When  $z = \exp \left[ j2\pi \ell \left[ \frac{L}{N} \right] \right]$ ,  $F(z) = F_{\ell}$ ; or, equivalently, when  $f = \left[ \frac{\ell L}{N\tau} \right]$ ,  $F(f) = F_{\ell}$ .

The inverse  $z$ -transform of (17) yields the impulse response of the digital filter in Figure 6.

$$\left\{ (F_0 \cdot \frac{L}{N}); n = 0, 1, 2, \dots, \frac{N}{L} - 1 \right\}$$

$$\left\{ (F_1 \cdot \frac{L}{N}) \cdot e^{\frac{j2\pi nL}{N}}; n = \frac{N}{L}, \frac{N}{L} + 1, \dots, \frac{2N}{L} - 1 \right\}$$

$$\left\{ (F_2 \cdot \frac{L}{N}) \cdot e^{\frac{j4\pi nL}{N}}; n = \frac{2N}{L}, \frac{2N}{L} + 1, \dots, \frac{3N}{L} - 1 \right\}$$

$$\vdots$$

$$\left\{ (F_{L-1} \cdot \frac{L}{N}) \cdot e^{\frac{j2\pi(L-1)Ln}{N}}; n = (L-1) \frac{N}{L}, (L-1) \frac{N}{L} + 1, \dots, N-1 \right\} \quad (19)$$

The magnitude and phase of this number sequence is illustrated in Figure 7. Those familiar with waveforms used in modern radar will recognize this filter response as that which is required to optimally process a Stepped Frequency Modulated (SFM) waveform (Ref. 8). As will be illustrated

in the following example, the filter of Figure 6 optimally processes, in a moving window fashion, the SFM waveform with many fewer multiplications than required by the convolution filter of Figure 1. Hence, Equation (17), which is realized as the filter of Figure 6, represents a method of filter synthesis which, for the SFM waveform, fulfills the goal set forth in the Introduction: to reduce the number of multiplications required in moving window processing. To illustrate this concept, consider the following example.

#### IV. Optimal Filtering of the Stepped Frequency Modulated Waveform

Suppose it is desired to synthesize a digital filter that is matched (in the signal processing sense) to the stepped frequency modulated waveform of Figure 8. In this figure,  $N$  and  $L$  are assumed to be 1024 and 16, respectively. The bandwidth of this waveform is  $\frac{1}{(4\tau)}$ , where  $\tau$  is the basic time delay of the filter. The magnitude of the frequency spectrum of this waveform is shown in Figure 9, and is repetitive with period  $\frac{1}{\tau}$ . The phase variation with frequency, which is not shown, is nearly quadratic.

The filter which is matched to this waveform is shown in Figure 10. Its impulse response is the conjugated time-inverse of the waveform shown in Figure 8. The filter's frequency response is the conjugate of the waveform spectrum shown in Figure 9, and is composed of  $16 \sin Nx / \sin x$  responses, each with unity weight. Figure 11 illustrates the composition of this filter spectrum. The ripples in the pass band and stopband are due to the sidelobes of the various  $\sin Nx / \sin x$  responses. Also note that nulls exist in the stopband of the filter where every  $\sin Nx / \sin x$  response has zero value.

The time response of this filter to the stepped frequency modulated (SFM) waveform of Figure 8 is shown in Figure 12. This response, which



is the convolution of the impulse response of the filter of Figure 10 and the SFM waveform illustrated in Figure 8, is very similar to a  $\sin x/x$  function in the vicinity of the mainlobe response with the accompanying high sidelobes. If it is desired to suppress these sidelobes, than amplitude weighting can be used. That is, the filter weights  $F_l; l = 0, 1, 2, \dots, L-1$  can be selected so that the time response is similar to that in Figure 12, but with lower sidelobes. As an example, suppose the  $\sin x/x$  sidelobes are to be lowered to 30 dB below the mainlobe. Sixteen weights which are obtained by sampling the Taylor Weight function (Ref. 9) can be used in the filter of Figure 10 to yield the time response of Figure 13.

The number of multiplications needed to process a sequence of samples with the filter of Figure 10 is  $L \cdot K$ , or  $16K$  where  $K$  is the number of samples to be processed. If the filter realization of Figure 3 had been used, the number of multiplications required would be  $2 \cdot N \cdot K$ , or  $2048K$ . Hence, the savings in multiplication for this particular example are a factor of 128. If nonunity weights are used with the filter of Figure 10, the number of multiplications required will be  $2 \cdot L \cdot K$  and the savings factor will be 64.

Of course, the filter matched to the waveform of Figure 8 could have been represented with fewer than 1024 points, since the waveform is oversampled by a factor of four. For example, the filter impulse response could have been represented by 256 points, in which case the resulting multiplication savings factor would be 32. However, the frequency response would look different than that shown in Figure 9. The waveform spectrum would be spread out over the repetition period so that the filter would be essentially an all-pass filter with an approximately quadratic phase function, and would have to be preceded with a linear-phase low-pass filter.

### V. Optimal Filtering of the Linearly Frequency Modulated Waveform

This technique can also be used to synthesize digital filters matched to other important waveforms. For example, of great importance to radar signal processing is the linearly frequency modulated (LFM) waveform (Ref. 8). The magnitude and phase characteristics of this waveform are very similar to those of the stepped frequency waveform illustrated in Figure 8. The big difference in the two is that the LFM waveform has a continuously quadratic phase function in time, whereas the SFM waveform has a piecewise continuous phase function that is only approximately quadratic. Because of the similarities of these two waveforms, one would expect that the filter of Figure 6 could be modified to optimally filter the LFM waveform.

If in the filter of Figure 6 there were more than one subfilter preceding each delay, as shown in Figure 14, then greater freedom in selecting the complex-valued impulse response would exist. In fact, if there were  $N/L$  subfilters preceding each delay, then any complex impulse response defined by (6) could be achieved by selecting the appropriate weights.

$$\left\{ F_{\ell, k}; \ell = 0, 1, 2, \dots, L-1; k = 0, 1, 2, \dots, \left[ \frac{N}{L} \right] - 1 \right\} \quad (20)$$

which are related to the impulse response  $\left\{ h(n); n = 0, 1, 2, \dots, N-1 \right\}$  as follows:

$$F_{\ell, k} = \sum_{n=\ell \cdot \frac{N}{L}}^{(\ell+1) \cdot \frac{N}{L} - 1} h(n) \exp \left[ \frac{-j2\pi kn}{N/L} \right]; \quad \begin{array}{l} \ell = 0, 1, 2, \dots, L-1 \\ k = 0, 1, 2, \dots, \left[ \frac{N}{L} \right] - 1 \end{array} \quad (21)$$

Hence, specific weights can be chosen so that the filter of Figure 14 is matched to the LFM waveform.

The number of multiplications required to process a sequence of input

samples with the filter of Figure 14 is  $K \cdot \left[ \frac{N}{L} \right]$  per subfilter for a total of  $2 \cdot K \cdot N$  for the composite filter (including the  $N$  frequency weights). This is the same number of multiplications as required by the convolution realization of Figure 3. Consequently, this realization has no advantage over that of Figure 3 if a filter perfectly matched to an LFM waveform is required. However, if a filter that is approximately matched to the LFM waveform is acceptable, then this synthesis approach is advantageous. For example, the filter of Figure 10 which only requires  $16 \cdot K$  multiplications is almost matched to an LFM waveform of length  $1024\tau$  and bandwidth  $1/(4\tau)$ . The phase function of the SFM filter is not exactly quadratic, as is desired for a filter matched to an LFM waveform. However, the mismatch is small, and for some applications the SFM filter of Figure 10 might be adequate for an LFM waveform.

If a filter which is slightly mismatched to an LFM waveform is acceptable, but less mismatch is desired than that provided by the SFM filter, then the filter of Figure 15 might be considered. In this filter, three subfilters per subsection are used. The complex weights can be chosen to approximate the LFM quadratic phase function, but with less mismatch than with one subfilter per subsection as in Figure 10. The number of multiplications needed for this filter is  $3 \cdot L \cdot K$ , or  $48 \cdot K$ . Hence, the savings in multiplications for this filter, as compared to the filter of Figure 7, are  $2048/48$ , or approximately 43.

It is obvious that the LFM quadratic phase function can be approximated with increasing accuracy by increasing the number of subfilters per subsection. Of course, the multiplications savings factor decreases with the increasing number of subfilters. The limit of this approach is when all  $(N/L)$  subfilters per subsection are used, in which case the filter is matched to

the LFM waveform and  $2 \cdot N \cdot K$  multiplications are required.

## VI. Conclusions

Nonrecursive digital filters can be synthesized by several techniques, as described by various authors. Furthermore, the complexity of the filter may depend on the synthesis technique chosen. This paper has presented a technique for designing some types of finite-duration impulse-response digital filters which require many fewer multiplications than other known techniques. Any desired finite-duration impulse response can be synthesized with this approach. However, the savings in multiplications depend on the impulse responses desired. For some well-known radar applications, such as pulse compression, the multiplication savings factor can be very large. The examples discussed in this paper indicate that the savings in the required multiplication rate can be factors of a hundred or more.

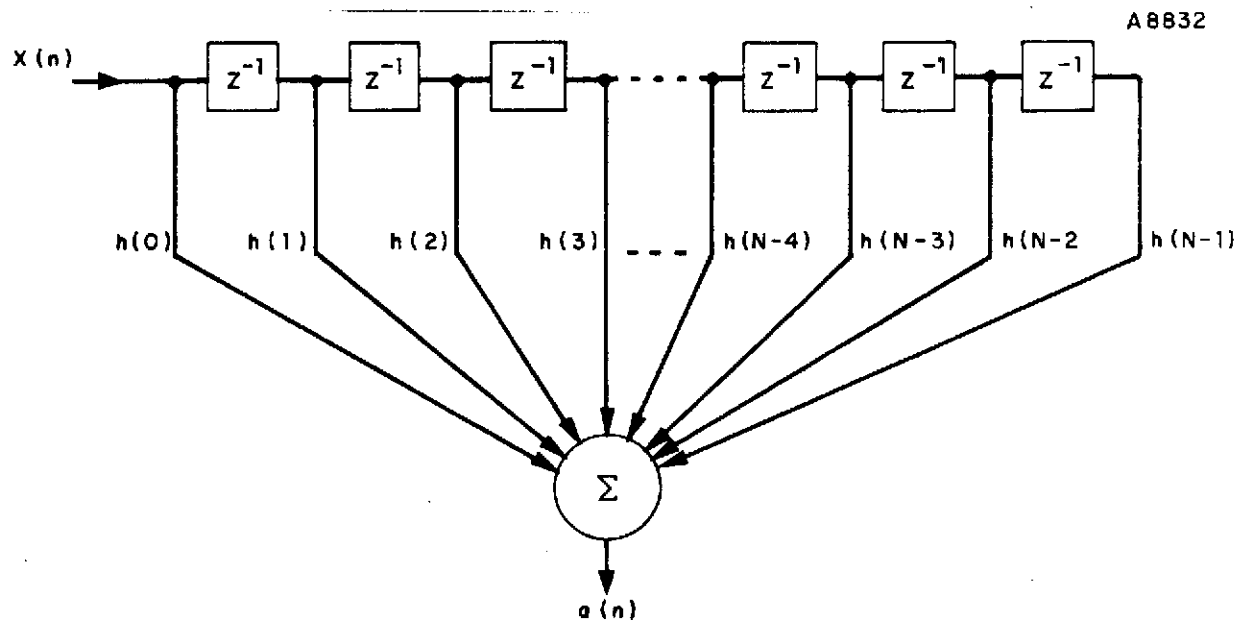


Figure 1. Convolution Realization of Nonrecursive Digital Filter

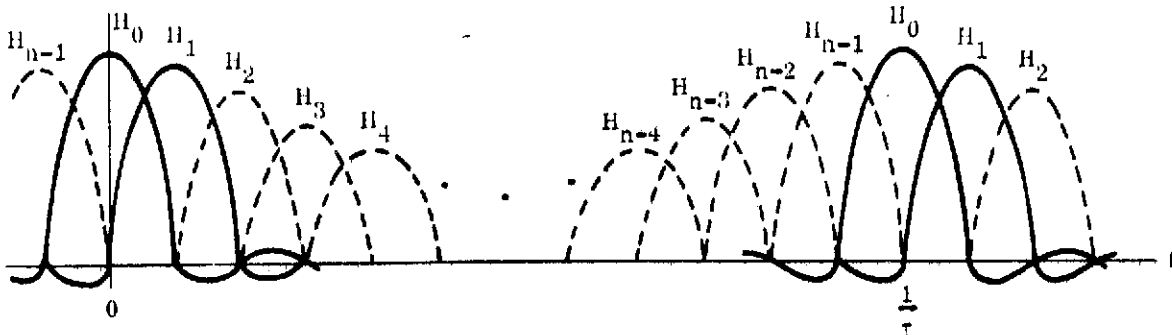


Figure 2. Composition of Filter Frequency Response

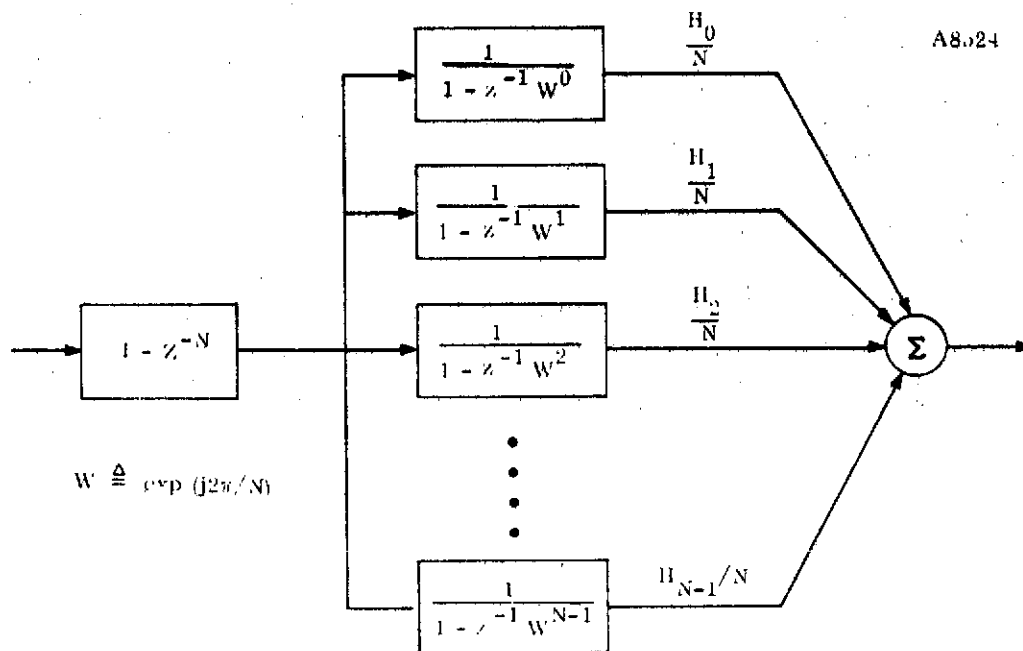
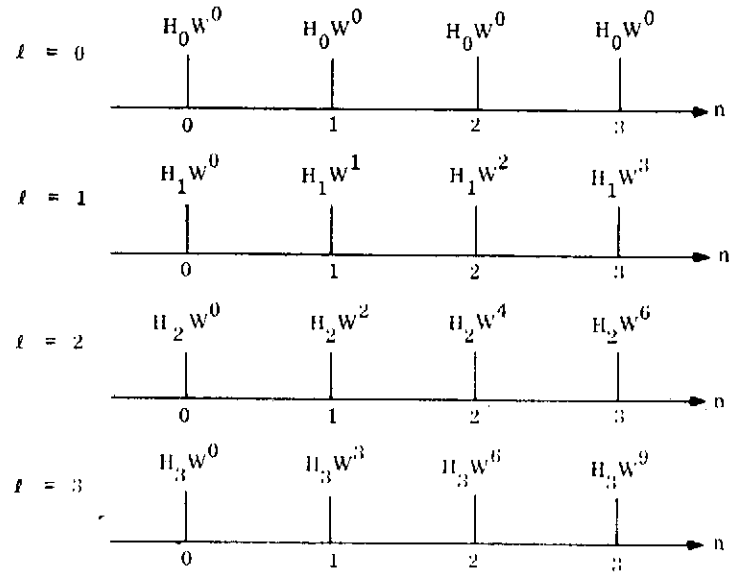
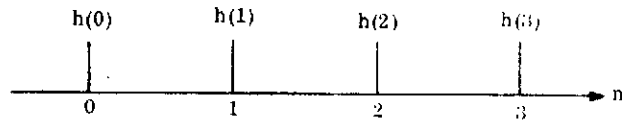


Figure 3. Frequency Response Realization



INDIVIDUAL SUBFILTER RESPONSES



COMPOSITE FILTER RESPONSE

$$h(0) = (H_0 W^0 + H_1 W^0 + H_2 W^0 + H_3 W^0)/4$$

$$h(1) = (H_0 W^0 + H_1 W^1 + H_2 W^2 + H_3 W^3)/4$$

$$h(2) = (H_0 W^0 + H_1 W^2 + H_2 W^4 + H_3 W^6)/4$$

$$h(3) = (H_0 W^0 + H_1 W^3 + H_2 W^6 + H_3 W^9)/4$$

$$W \triangleq \exp(j2\pi/4)$$

Figure 4. Composition of Digital Filter Impulse Response ( $N = 4$ )

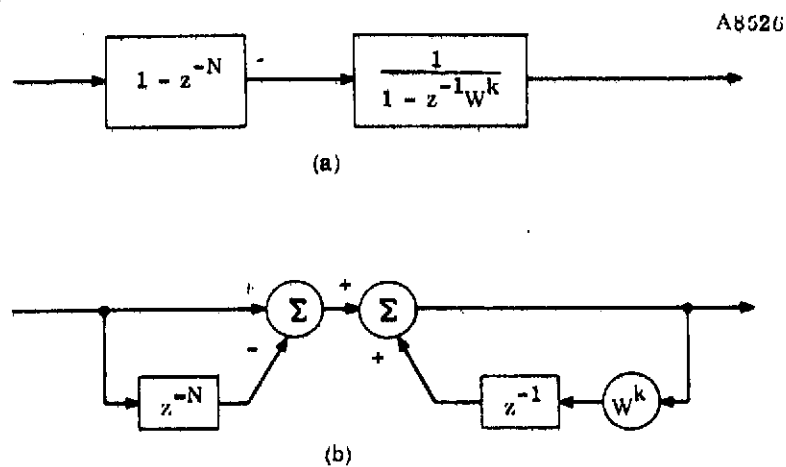


Figure 5. Digital Filter Consisting of One Subfilter

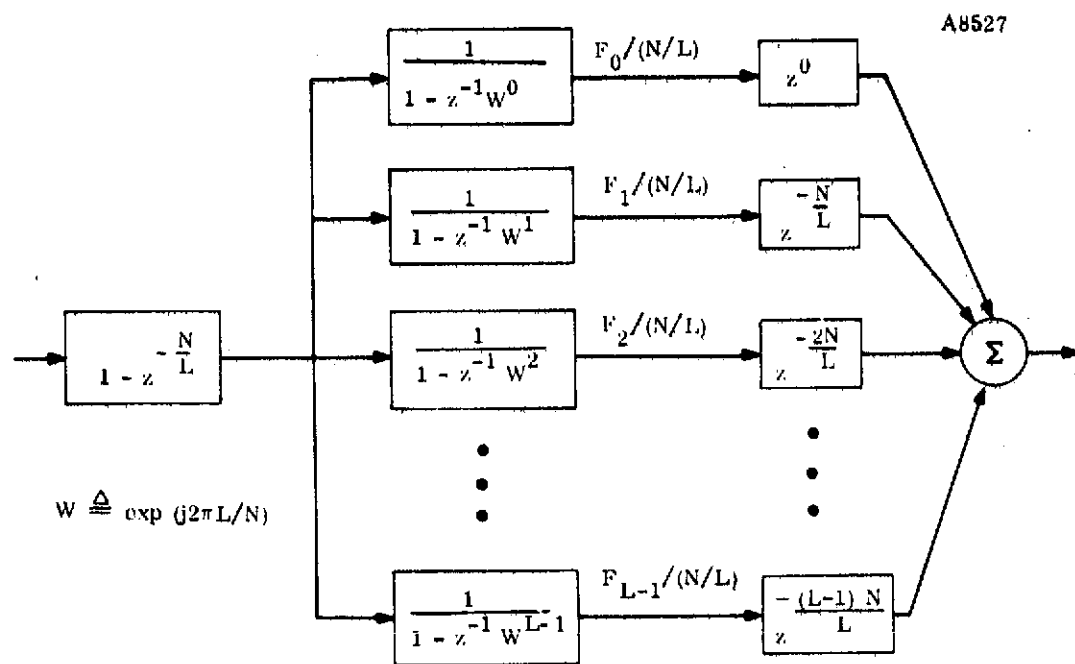


Figure 6. A Specialized Digital Filter Realization



A8528

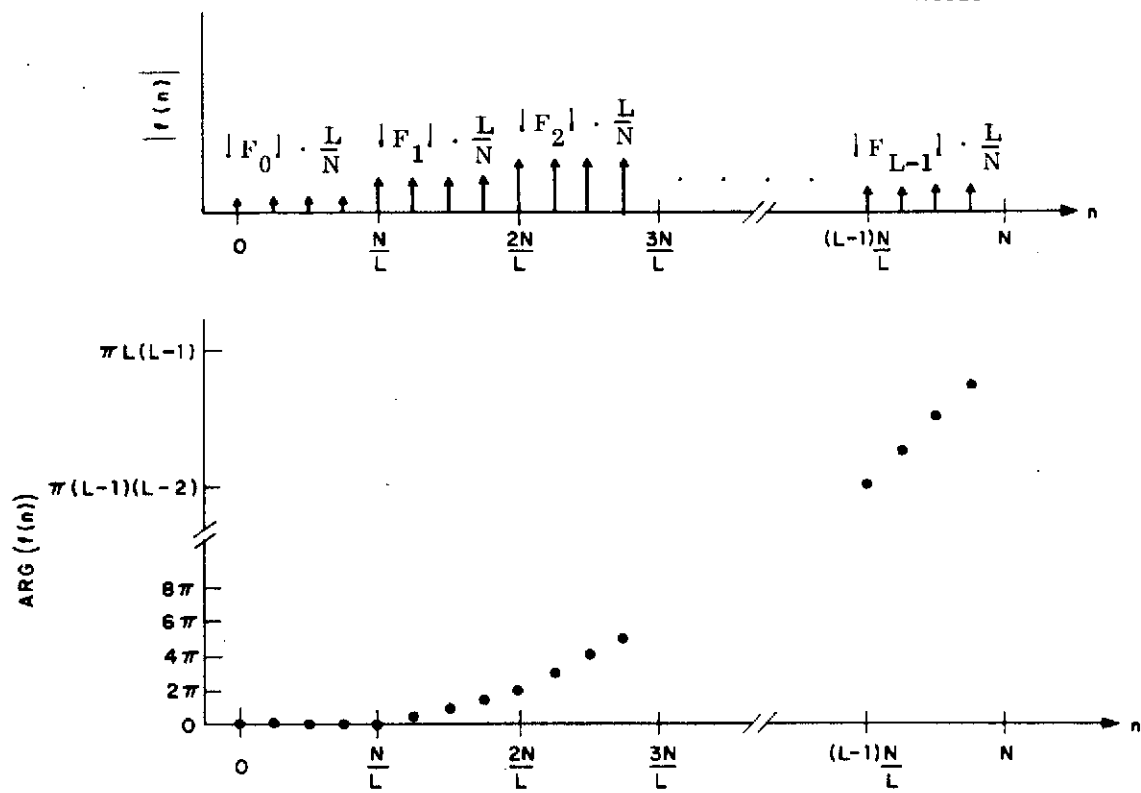


Figure 7. The Impulse Response of the Filter in Figure 6

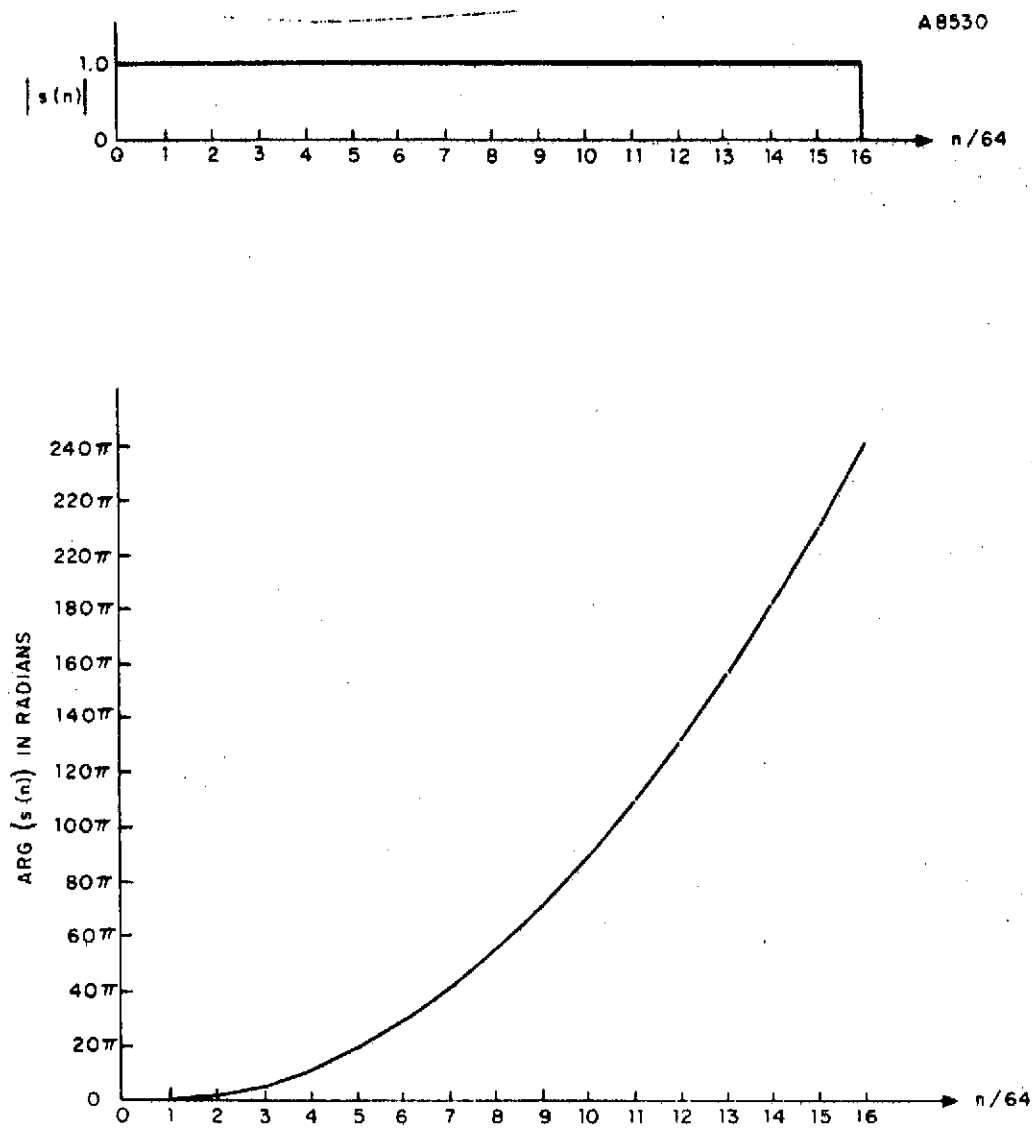


Figure 8. A Sampled Stepped Frequency Modulated Waveform ( $N = 1024$ )

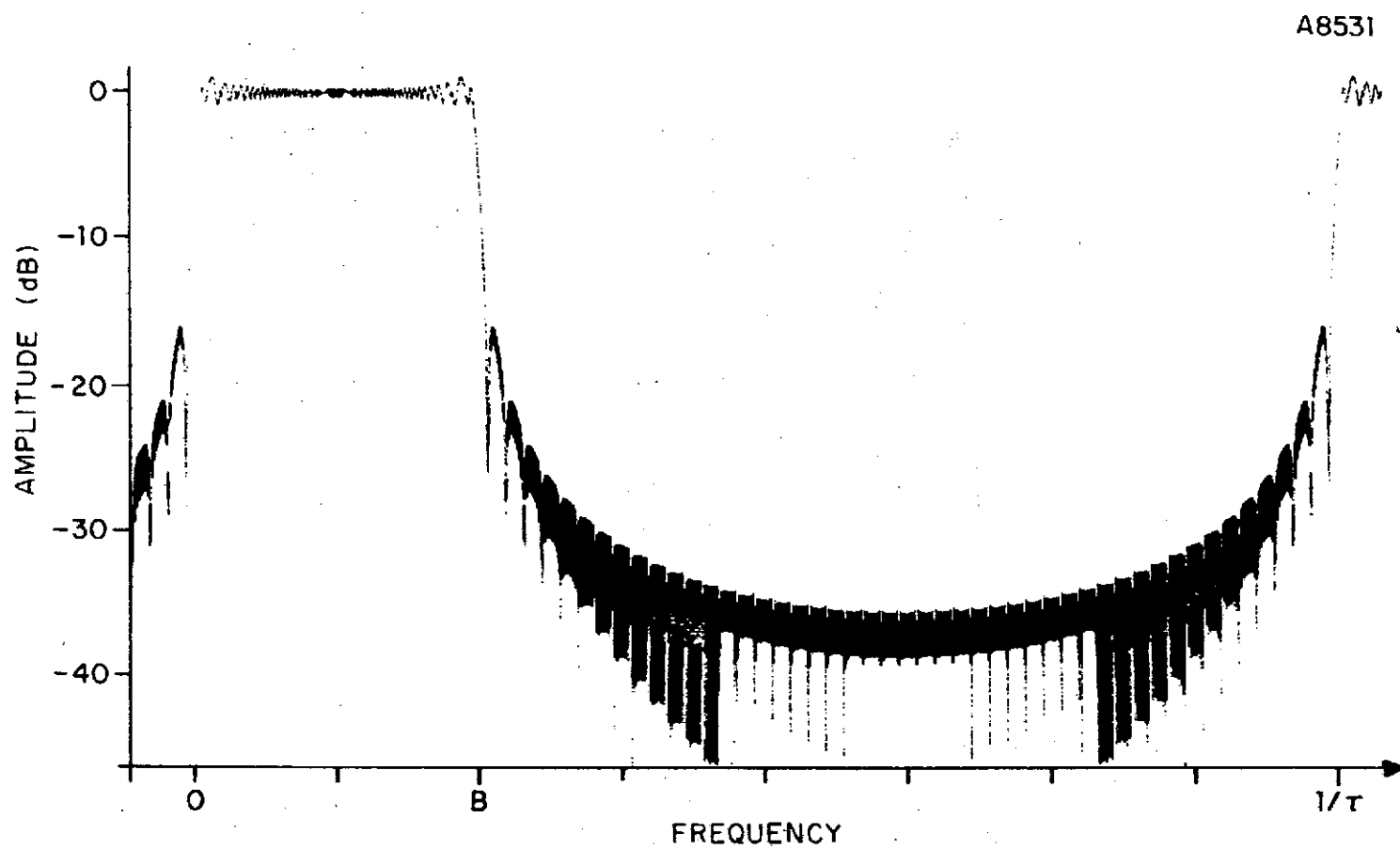


Figure 9. Magnitude of Frequency Response of the Sampled Stepped Frequency Waveform

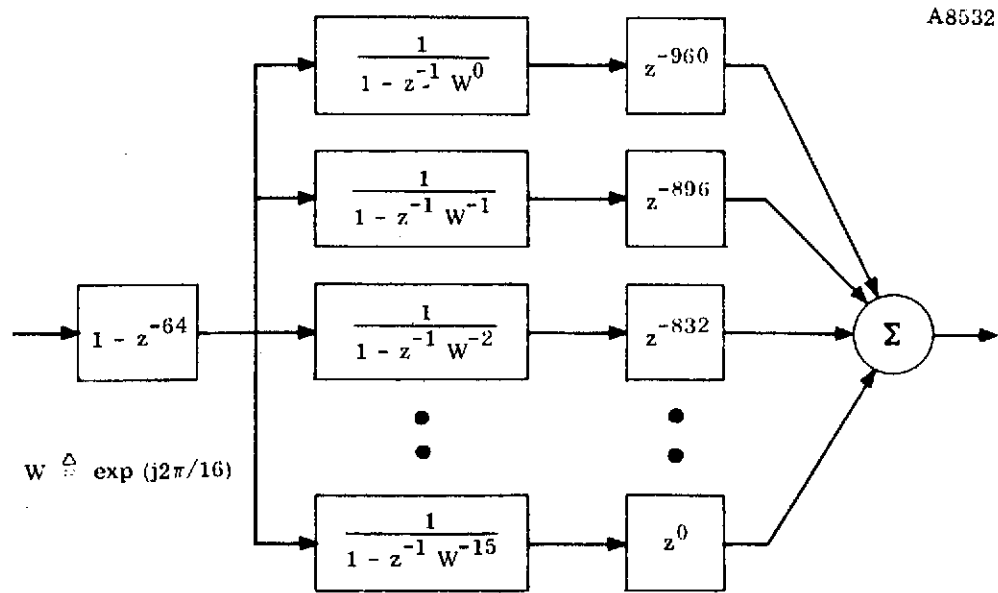


Figure 10. Digital Filter Matched to the Sampled Stepped Frequency Waveform

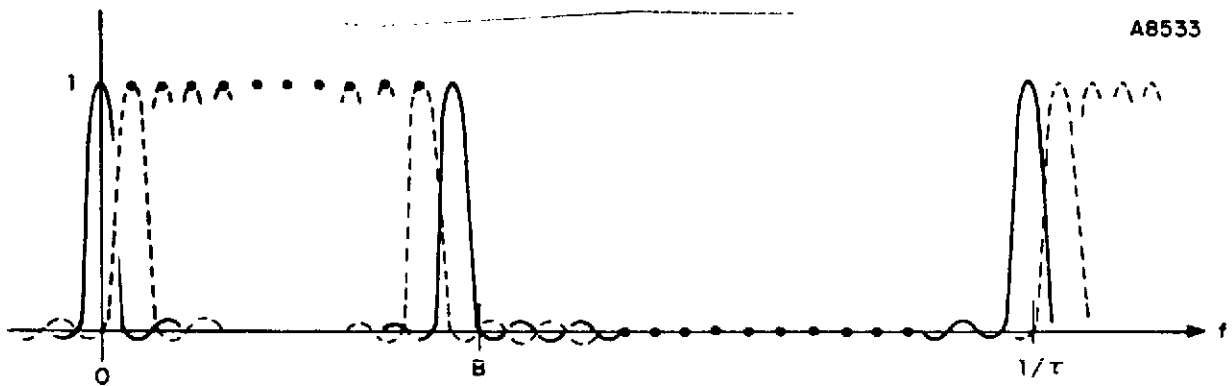


Figure 11. Composition of the Frequency Spectrum of Figure 9

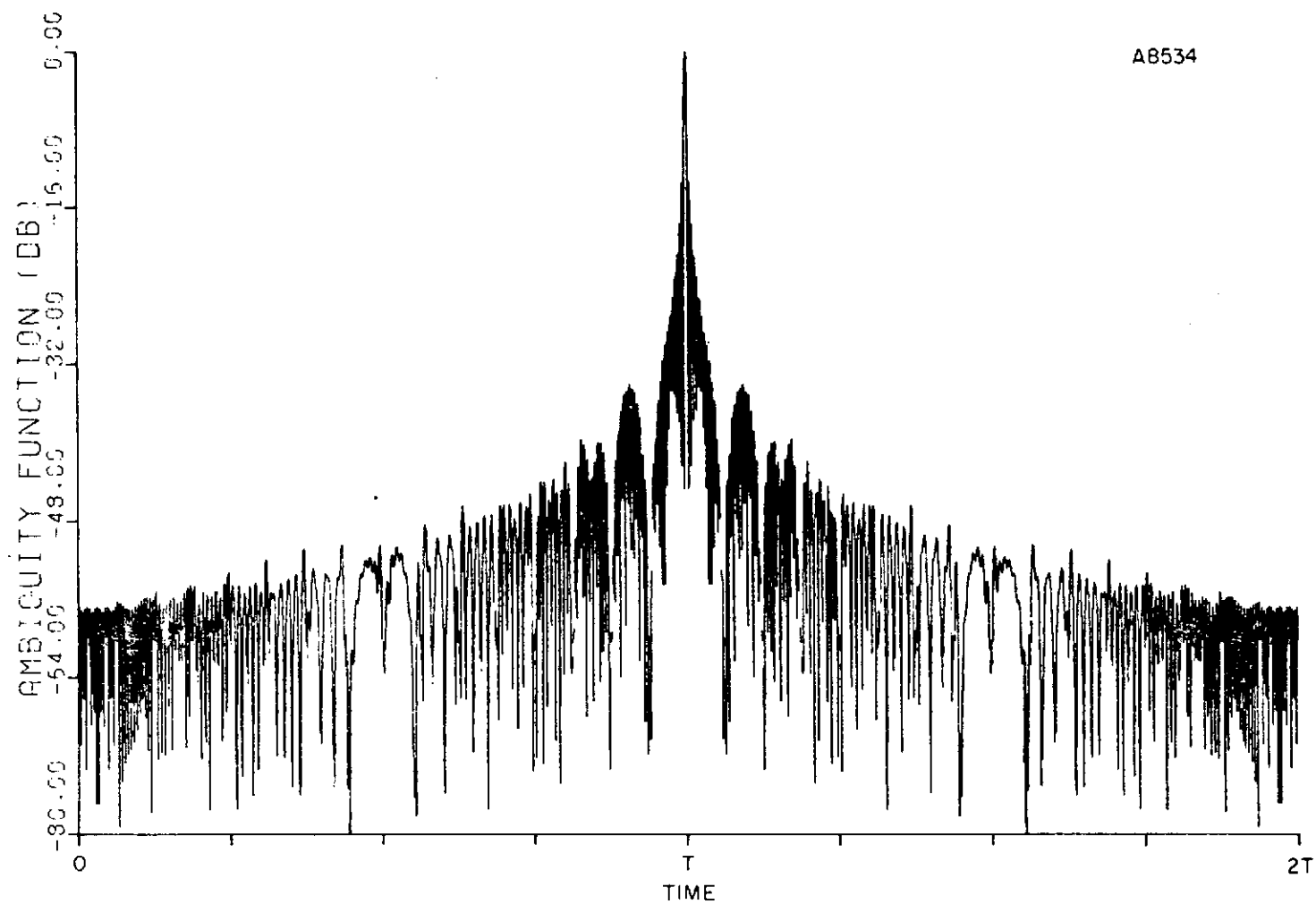


Figure 12. Time Response of Digital Filter to Step Frequency Modulated Waveform  
Uniform Amplitude Weighting

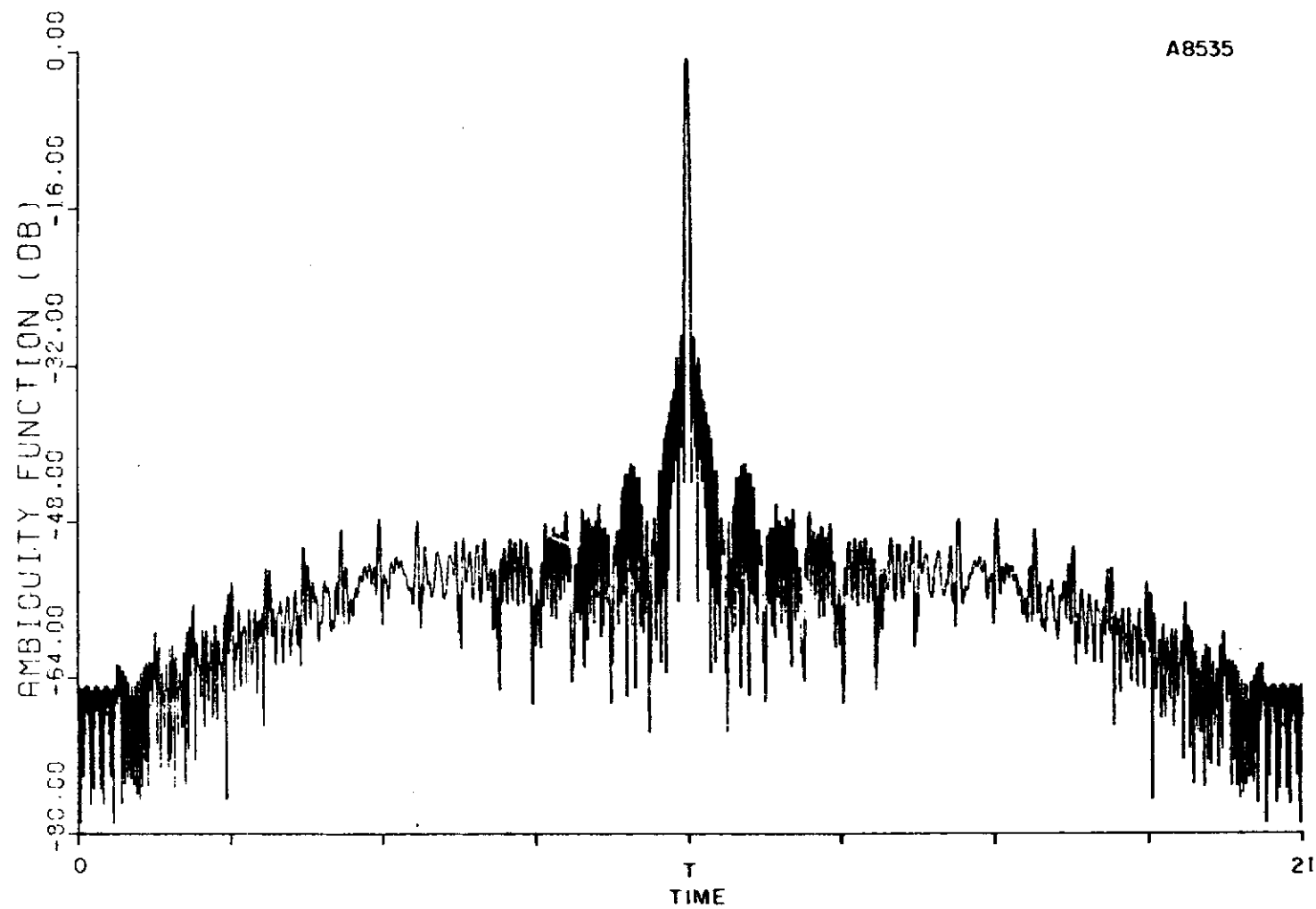
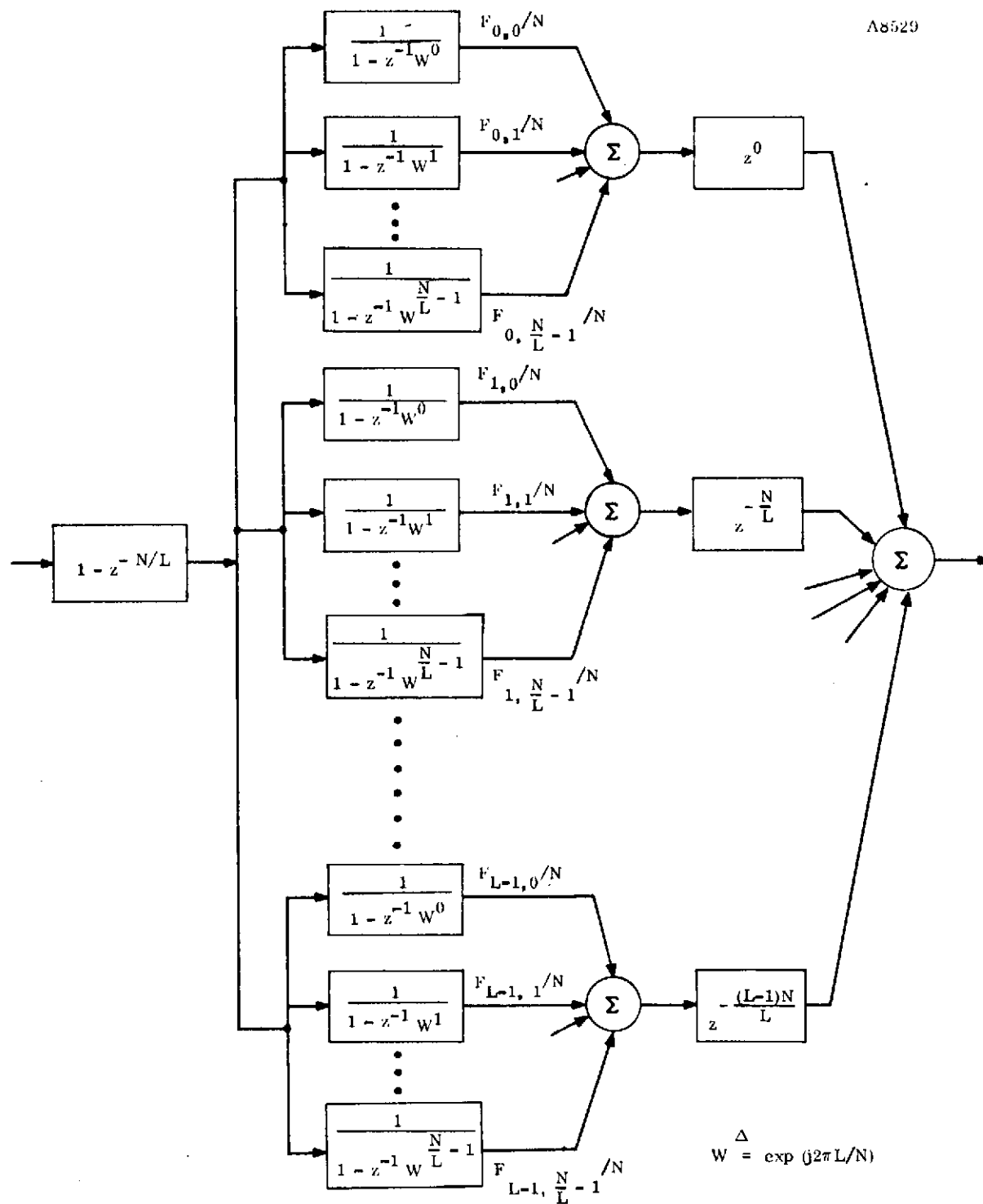


Figure 13. Time Response of Digital Filter to Step Frequency Modulated Waveform  
Taylor Amplitude Weighting (-30 dB,  $n = 6$ )

Figure 14. A Digital Filter Realization with  $L$  Subsections

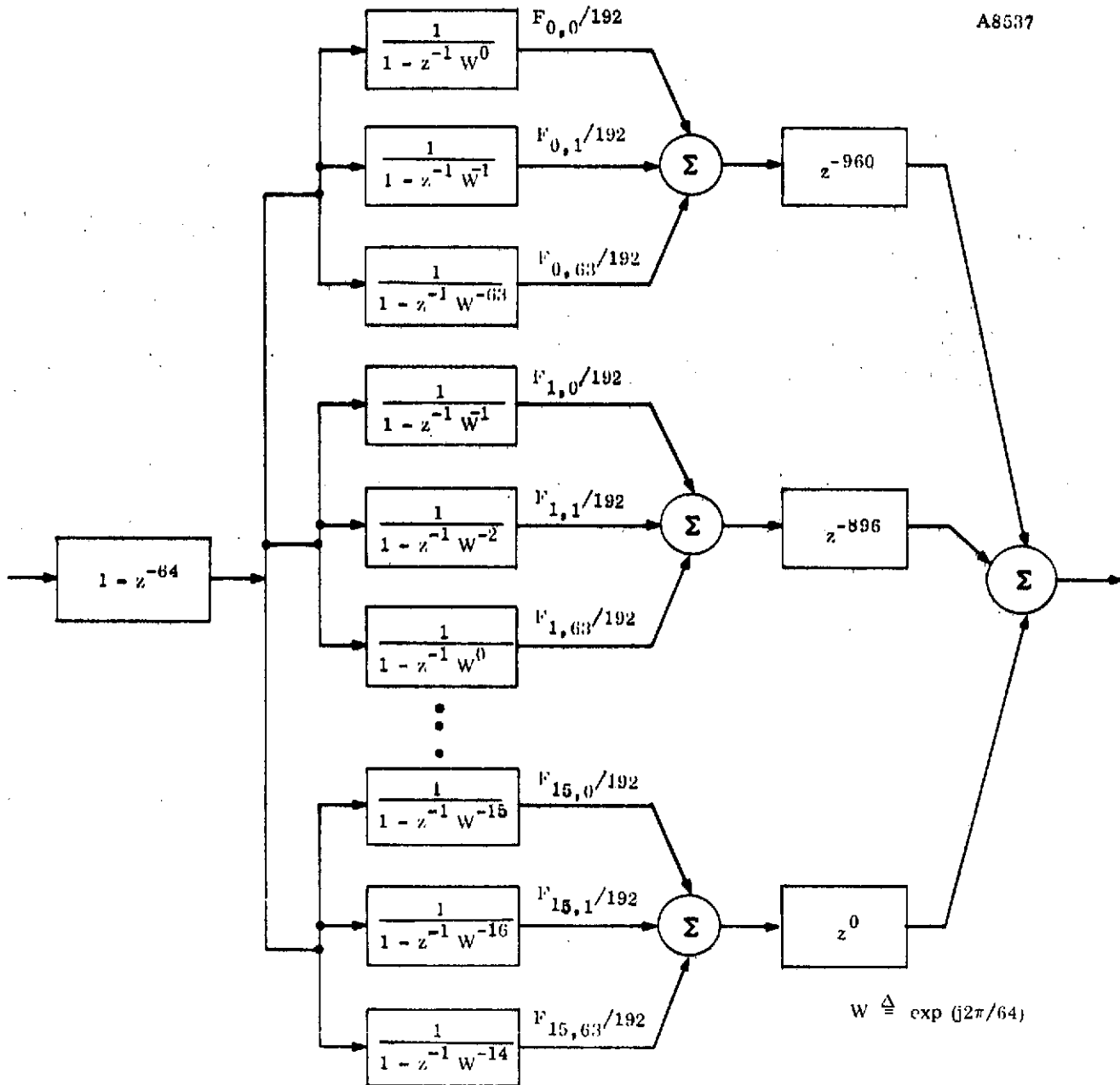


Figure 15. Digital Filter Approximately Matched to the Linearly Frequency Modulated Waveform



REFERENCES

1. Gold, B., and Rader, C.M., Digital Processing of Signals, McGraw-Hill Book Co., New York, New York, 1969.
2. Helms, H. D., "Nonrecursive Digital Filters: Design Methods for Achieving Specifications on Frequency Response", IEEE Transactions on Audio and Electroacoustics, Vol. AV-16, September 1968.
3. Gold, B., "Signal Processing Hardware for General Purpose Computers", NEREM Record, 1969.
4. Rabiner, L.R., Gold, B., and McGonegal, C.A., "An Approximation Problem for Nonrecursive Digital Filters", IEEE Transactions on Audio and Electroacoustics, Vol. AU-18, June 1970.
5. Rader, C.M., and Gold, B., "Digital Filter Design Techniques in the Frequency Domain", Proc. IEEE 55, 2, 149, 1967.
6. Van Trees, H.L., Detection, Estimation, And Modulation Theory - Part I, John Wiley and Sons, Inc., New York, NY, 1968 .
7. Brigham, E.O., and Morrow, R.E., "The Fast Fourier Transform", IEEE Spectrum, December 1967.
8. Cook, C.E., and Bernfeld, M., Radar Signals - Introduction to Theory and Application, Academic Press, 1967 .
9. Taylor, T.T. , "Design of Line-Source Antennas for Narrow Beamwidth and Low Side Lobes", IRE Transactions on Antennas and Propagation, January 1955.

#### IV. ON THE OUTPUT OF MULTIPLE SERVER QUEUES WHERE NOT ALL SERVERS ARE IDENTICAL

It is known that if the input to an  $N$  server queue, with the servers having exponential service time distributions with identical means  $1/\sigma$ , is Poisson with parameter  $\lambda$ , then, after attaining equilibrium, the output is also Poisson with parameter  $\lambda$ . It is shown in this paper that as long as the servers have exponential service time distributions, even if not identical means, and equilibrium is attained with a Poisson input of parameter  $\lambda$ , the output will also be Poisson with parameter  $\lambda$ . Furthermore, for the output of each individual server to be Poisson, it is necessary and sufficient that the probability of a particular server being busy, conditioned on the number of customers in the system, be equal to the probability of any other server being busy, under the same conditioning. The output from server  $i$  will then have parameter  $\lambda \frac{\sigma_i}{\sigma}$  and be independent of the output from the other servers. However, for such equiprobable conditions to exist, restrictions must be placed on the dispersion of the mean service times of servers.

##### I. Problem Description

The problem of describing the output of a queue has been given considerable attention, primarily due to the problem's application to tandem queues. However, in most of the literature that attention has been focused on queues where the  $N$  servers have identical service time distributions and on the output of the total system rather than on the output of any individual server. In this paper the output of an individual server

is examined. The motivation for this examination is the consideration of tandem queues in networks where the output of each server may feed into a different queue than the queue fed by the output of another server. In particular, modeling of communication networks for message transmission from a terminal to a set of processors over channels of fixed capacity may be posed as an arrangement of tandem queues where the channel capacities and message lengths determine the various service times of the servers.

The general queueing system for this paper is shown in Figure 1. The input is Poisson with parameter  $\lambda$ . Each server  $i$  has an exponential service time distribution with parameter  $\sigma_i$ , not necessarily all the same. P. J.

Burke<sup>1</sup> has shown that if the service time distributions are all identical and

$$\sum_{i=1}^N \sigma_i = N\sigma > \lambda$$

and equilibrium has been attained, the output of the overall system will be Poisson with parameter  $\lambda$ , same as the input. Edgar Reich<sup>2</sup> has also shown this result by arguments different than those of Burke. In this paper the basic strategy of Burke is followed to show that the output from the overall system is Poisson even if the servers are not identically distributed, so long as equilibrium is attained and the system is Markovian. Furthermore, for the output of each individual server to be Poisson, it is necessary and sufficient that the probability of a server being busy, conditioned on the number of customers in the system, be equal to the probability of any other server being busy, under the same conditioning. The output from a particular server  $i$  will then have parameter  $\lambda \frac{\sigma_i}{\sigma}$  and be independent of the output from the other servers. However, for such equiprobable conditions to exist, restrictions must be placed on the dispersion of the mean service times.

## II. Equilibrium Probabilities

In order to pursue the analysis it is first necessary to determine the equilibrium probabilities describing the queue. The probability of a particular server  $j$  being busy conditioned on  $i$  customers in the system will be represented as

$$Q_j^i = \Pr (\text{customer in } \sigma_j \mid i \text{ customers in system})$$

The mean service time of the system, conditioned on  $i$  customers in the system, is defined as

$$\mu_i = \sum_{j=1}^N \sigma_j Q_j^i$$

Finally, define the equilibrium probability of the state of the system as

$$P_i = \Pr (i \text{ customers in system})$$

The equilibrium relations describing a system of  $N$  servers may now be written as below.

$$\begin{aligned} \lambda P_0 &= \mu_1 P_1 \\ &\vdots \\ (\lambda + \mu_i) P_i &= \lambda P_{i-1} + \mu_{i+1} P_{i+1} & 1 \leq i \leq N-2 \\ &\vdots \\ (1) \quad (\lambda + \mu_{N-1}) P_{N-1} &= \lambda P_{N-2} + \sigma P_N \\ &\vdots \\ (\lambda + \sigma) P_j &= \lambda P_{j-1} + \sigma P_{j+1} & N \leq j \leq \infty \\ &\vdots \end{aligned}$$

These equations are subject to the constraint

$$(2) \quad \sum_{i=0}^{\infty} P_i = 1$$

Define  $\xi = \frac{\lambda}{\sigma}$ ,  $\xi_i = \frac{\lambda}{\mu_i}$ ,  $1 \leq i \leq N-1$ , and  $\xi_0 = 1$ . The solution to the above equations (1) subject to the constraint (2) is then easily found to be

$$P_j = \prod_{i=0}^j \xi_i P_0 \quad 1 \leq j \leq N-1$$

$$(3) P_j = (\xi)^{j+1-N} \prod_{i=0}^{N-1} \xi_i P_0 \quad N \leq j \leq \infty$$

$$P_0 = \frac{1 - \xi}{\left( \sum_{j=0}^{N-1} \prod_{i=0}^j \xi_i \right) - \xi \left( \sum_{j=0}^{N-2} \prod_{i=0}^j \xi_i \right)}$$

### III. Output of General Queue

The equilibrium probabilities may be interpreted as the probabilities of finding the system in a particular state at any randomly selected instant of time. Central to the developments in this paper is the result that the probabilities of finding the system in a particular state at a randomly selected instant of time drawn from the set of instants immediately after departures are also equilibrium probabilities. This result is stated below as a lemma.

#### Lemma 1

The state of the general queue, subject to Poisson input and having attained equilibrium, after a departure is described by the equilibrium probabilities.

With  $\mu_i$  replacing  $\tau$  for  $1 \leq i \leq N-1$  the proof of the lemma is completely analogous to the demonstration of a similar lemma by Burke.

The above lemma and replacements can be used to demonstrate the following theorem which is a simple extension of Burke's theorem.

Theorem 1

The output of the general queue with Poisson input having parameter  $\lambda$ , and after attaining equilibrium, is also Poisson with parameter  $\lambda$ .

The demonstration is again completely analogous to the demonstration by Burke.

IV. Output of Individual Server

The principal theorem of this paper pertains to the output of the individual servers in an N server queue. The theorem is stated below.

Theorem 2

For the N server queue with all servers having exponential service time distributions, but not necessarily identical means, and having a Poisson input with parameter  $\lambda$ , and having attained equilibrium, the output of each individual server will be Poisson if and only if it is equiprobable that a customer be in server i as in server j, for any i, j, conditioned on 1 customers in the system. The Poisson outputs will each be independent of all the other outputs and will have parameter  $\lambda \frac{\sigma_i}{\sigma}$ .

The proof of this theorem will be presented in two parts, the first part showing sufficiency and the second part showing necessity.

For ease of notation and without loss of generality, when referring to an individual server, server one with exponential service time parameter  $\sigma_1$  will be chosen. It will be easily seen that extension of what formulas are based on referring to an individual server from server one to any other server i is only a matter of notational complexity.

The sufficiency part of the proof follows very closely to Burke's proof. <sup>(1)</sup>

---

<sup>(1)</sup> Consequently, only the significant deviations from the work of Burke will be given.

The necessity part of the proof will be more involved. The general solution of the system of equations will be expanded as a matrix exponential, and in order to have a Poisson output various summation equalities will result. These equalities will then be shown to necessitate the conditions of the theorem.

#### V. Sufficiency

The notation of Burke is redefined to pertain to an individual server rather than the general system. Let  $t_0$  be any randomly selected instant of time, (later taken to be zero for convenience.) Let  $L$  be the length of time from  $t_0$  to the instant of the next departure from  $\sigma_1$ , and define

$$F_k(t) = \Pr [ n(t) = k, L > t - t_0 ]$$

Note

$$\sum_{k=0}^{\infty} F_k(t) = F(t) = \Pr (L > t - t_0)$$

and

$$F_k(t_0) = P_k .$$

A new symbol, interpreted as the portion of the conditional mean service time  $\mu_i$  contributed by the server  $j$ , is defined as

$$\eta_j^i = \sigma_j Q_j^i$$

Note

$$\sum_{j=1}^N \eta_j^i = \mu_i$$

Before proceeding with the sufficiency part of the proof a lemma is given. The lemma is quite comparable to the lemma stated earlier.

#### Lemma 2

If it is equiprobable that a customer be in server  $i$  as in server  $j$ , for

any  $i, j$ , conditioned on 1 customers in the system, then the probabilities describing the state at any random time selected from the set of departure instants of server  $i$  are the equilibrium probabilities.

The conditions of the lemma are simply

$$Q_j^i = Q_k^i \quad \forall i, j, k$$

Thus

$$\frac{\eta_j^i}{\mu_i} = \frac{\sigma_j Q_j^i}{\sum_{j=1}^N \sigma_j Q_j^i} = \frac{\sigma_j Q_j^i}{Q_j^i \sigma} = \frac{\sigma_j}{\sigma}$$

The following equations describe the system.

$$\begin{aligned} F_0(t+dt) &= F_0(t)(1-\lambda dt) + F_1(t)(1-\lambda dt)(\mu_1 - \eta_1^1) dt \\ &\vdots \\ F_i(t+dt) &= F_{i-1}(t)(\lambda dt)(1-\mu_{i-1} dt) + F_i(t)(1-\lambda dt)(1-\mu_i dt) \\ &\quad + F_{i+1}(t)(1-\lambda dt) [ (\mu_{i+1} - \eta_{i+1}^{i-1}) dt ] \\ &\vdots \\ F_{N-1}(t+dt) &= F_{N-2}(t)(\lambda dt)(1-\mu_{N-2} dt) + F_{N-1}(t)(1-\lambda dt)(1-\mu_{N-1} dt) \\ (7) \quad &\quad + F_N(t)(1-\lambda dt) [ (\sigma - \sigma_1) dt ] \\ F_N(t+dt) &= F_{N-1}(t)(\lambda dt)(1-\mu_{N-1} dt) + F_N(t)(1-\lambda dt)(1-\sigma dt) \\ &\quad + F_{N+1}(t)(1-\lambda dt) [ (\sigma - \sigma_1) dt ] \\ &\vdots \\ F_j(t+dt) &= F_{j-1}(t)(\lambda dt)(1-\sigma dt) + F_j(t)(1-\lambda dt)(1-\sigma dt) \end{aligned}$$



$$\begin{aligned}
& + F_{j+1}(t)(1-\lambda dt)[(\sigma - \sigma_1) dt] \\
& \vdots \quad N+1 \leq j < \infty.
\end{aligned}$$

Performing the usual manipulations, these equations become

$$\begin{aligned}
(8) \quad & F'_0 = -\lambda F_0 + (\mu_1 - \eta_1^1) F_1 \\
& \vdots \\
& F'_i = \lambda F_{i-1} - (\lambda + \mu_i) F_i + (\mu_{i+1} - \eta_1^{i+1}) F_{i+1} \quad 1 \leq i \leq N-2 \\
& \vdots \\
& F'_{N-1} = \lambda F_{N-2} - (\lambda + \mu_{N-1}) F_{N-1} + (\sigma - \sigma_1) F_N \\
& \vdots \\
& F'_j = \lambda F_{j-1} - (\lambda + \sigma) F_j + (\sigma - \sigma_1) F_{j+1} \quad N \leq j < \infty
\end{aligned}$$

This is a system of first order, linear, homogeneous differential equations subject to the initial conditions  $F_k(0) = P_k$ , where  $t_0$  has been chosen as zero for convenience. The solution of this system of equations is unique and it can be easily verified that with  $Q_j^i = Q_k^i$  the solution is

$$F_k(t) = P_k e^{-\lambda \frac{\sigma_i}{\sigma} t}$$

The remainder of the proof of the lemma is quite analogous to Burke's demonstrations.

To proceed with the sufficiency part of the theorem's proof some symbols are redefined. Let  $L$  be the length of time from any departure instant of  $\sigma_1$ , i.e., the interdeparture interval length of  $\sigma_1$ , and  $n(t)$  be the state at time  $t$  after some arbitrarily chosen departure instant of  $\sigma_1$ ,  $t_d$ , where for convenience  $t_d$  is taken as zero. Define

$$F_k(t) = \Pr [ n(t) = k, L > t ]$$

and note

$$\sum_{k=0}^{\infty} F_k(t) = F(t) = \Pr (L > t)$$

With this notation the set of differential equations (8) can be seen to describe the system, and with  $Q_j^i = Q_k^i$  the solution to the equations is

$$F_k(t) = P_k e^{-\lambda \frac{\sigma_i}{\sigma} t}.$$

Following the format of Burke this is easily used to show that if  $Q_j^i = Q_k^i$ , for all  $i, j, k$ , the output is Poisson with parameter  $\lambda \frac{\sigma_i}{\sigma}$ .

Finally note that by theorem one the overall output is Poisson with parameter  $\lambda$  and hence has the characteristic function

$$\Phi(s) = e^{-\lambda + \lambda s}$$

From the results of this section, the characteristic function of the  $i^{\text{th}}$  output is

$$\Phi_i(s) = e^{-\lambda \frac{\sigma_i}{\sigma} + \lambda \frac{\sigma_i}{\sigma} s}$$

$$\begin{aligned} \text{With} \\ \prod_{i=1}^N \Phi_i(s) &= e^{-\lambda \sum_{i=1}^N \frac{\sigma_i}{\sigma} + \lambda \sum_{i=1}^N \frac{\sigma_i}{\sigma} s} \\ &= e^{-\lambda + \lambda s} \\ &= \Phi(s) \end{aligned}$$

it is easy to see from the convolution theorem that all of the outputs are independent of each other.

#### Necessity

To prove the necessity part of the theorem, it will be assumed that the output from the server is Poisson with parameter  $\beta_i$ .

The differential equations describing the system may be written in matrix notation as

$$(10) \quad \underline{F}' = A\underline{F}$$

with

$$A = \begin{bmatrix} -\lambda & (\mu_1 - \eta_1) & & & \\ \lambda & -(2 + \mu_1) & (\mu_2 - \eta_1^2) & & \\ 0 & \lambda & -(\lambda + \mu_2) & (\mu_3 - \eta_1^3) & \\ & \ddots & & & \\ 0 & 0 & \lambda & -(\lambda + \mu_k) & (\mu_{k+1} - \eta_1^{k+1}) \\ & \ddots & & & \\ 0 & 0 & 0 & \lambda & -(\lambda + \sigma) & (r - r_1) \\ & \ddots & & & & \end{bmatrix}$$

The system of differential equations is subject to the initial conditions

$$\underline{F}(0) = \underline{P}$$

where  $t_0$  is a randomly selected instant of time conveniently taken as zero and  $\underline{P}$  is the equilibrium probability vector

$$\underline{P} = \begin{bmatrix} P_0 \\ P_1 \\ \vdots \end{bmatrix}$$

It is well known that the solution to these equations may be written as

$$\begin{aligned} \underline{F} &= e^{At} \underline{P} \\ &= \left( I + At + A^2 \frac{t^2}{2!} + \dots \right) \cdot \underline{P} \\ &= \underline{P} + A\underline{P}t + A^2 \underline{P} \frac{t^2}{2!} + \dots \end{aligned}$$

The product  $A\underline{P}$  can be shown to be

$$\underline{AP} = \begin{bmatrix} -\lambda \frac{\eta_1}{\mu_1} P_0 \\ \vdots \\ -\lambda \frac{\eta_1}{\mu_{\ell+1}} P_{\ell} \\ \vdots \\ -\lambda \frac{\sigma_1}{\sigma} P_{N-1} \\ \vdots \\ -\lambda \frac{\sigma_1}{\sigma} P_k \\ \vdots \end{bmatrix}$$

Since

$$\sum_k F_k(t) = \Pr(L > t)$$

and the output is Poisson, implying

$$\begin{aligned} \Pr(L > t) &= e^{-\beta_1 t} \\ &= 1 - \beta_1 t + \beta_1^2 \frac{t^2}{2!} - \dots \end{aligned}$$

Therefore,

$$\sum_k \underline{AP} = \beta_1$$

and thus it can be shown that

$$\beta_1 = \lambda \frac{\sigma_1}{\sigma} + \lambda \sum_{k=0}^{N-2} \left( \frac{\eta_1}{\mu_{k+1}} - \frac{\sigma_1}{\sigma} \right) P_k$$

As before, the characteristic function of the overall output is

$$\Phi(s) = e^{-\lambda + \lambda s}$$

and the characteristic function of the output of each individual server is

$$\Phi_i(s) = e^{-\left[ \lambda \frac{\sigma_i}{\sigma} + \lambda \sum_{k=0}^{N-2} \left( \frac{\eta_i^{k+1}}{\mu_{k+1}} - \frac{\sigma_i}{\sigma} \right) P_k \right]} + \left[ \lambda \frac{\sigma_i}{\sigma} + \lambda \sum_{k=0}^{N-2} \left( \frac{\eta_i^{k+1}}{\mu_{k+1}} - \frac{\sigma_i}{\sigma} \right) P_k \right] s$$

Thus again the product is found to be

$$\prod_{i=1}^N \Phi_i(s) = e^{-\lambda + \lambda s} = \Phi(s)$$

and therefore the individual outputs must be independent of all the other outputs.

Having established the above result, the remaining portion of the proof will proceed under the assumption that the individual outputs are each Poisson and independent of the other outputs.

Combining previous results with the independence assumption gives

$$\prod_{i=1}^N P[L_i > t | n(t) = k] = e^{-\lambda t} = 1 - \lambda t + \lambda^2 \frac{t^2}{2!} + O(t^3 +)$$

where by  $O(t^3 +)$  is meant terms involving the third or higher powers of  $t$ .

Now

$$[Pr L_i > t | n(t) = k] = \frac{P[L_i > t, n(t) = k]}{Pr[n(t) = k]}$$

and it can be shown that

$$A^2 \underline{P} = \begin{bmatrix} \lambda^2 \frac{\eta_1^1 \eta_1^2}{\mu_1 \mu_2} P_0 + \lambda^2 \left( \frac{\eta_1^1}{\mu_1} - \frac{\eta_1^2}{\mu_2} \right) P_0 \\ \vdots \\ \lambda^2 \frac{\eta_1^{\ell+1} \eta_1^{\ell+2}}{\mu_{\ell+1} \mu_{\ell+2}} P_e + \lambda^2 \left( \frac{\eta_1^{\ell+1}}{\mu_{\ell+1}} - \frac{\eta_1^{\ell+2}}{\mu_{\ell+2}} \right) P_e + \lambda \mu_{\ell} \left( \frac{\eta_1^{\ell+1}}{\mu_{\ell+1}} - \frac{\eta_1^{\ell}}{\mu_{\ell}} \right) P_e \\ \vdots \\ \lambda^2 \frac{\sigma_1^2}{\sigma^2} P_{N-1} + \lambda \mu_{N-1} \left( \frac{\sigma_1}{\sigma} - \frac{\eta_1^{N-1}}{\mu_{N-1}} \right) P_{N-1} \\ \lambda^2 \frac{\sigma_1^2}{\sigma^2} P_N \\ \vdots \\ \lambda^2 \frac{\sigma_1^2}{\sigma^2} P_k \\ \vdots \end{bmatrix}$$

$A \underline{P}$  and  $A^2 \underline{P}$  can be used to form a matrix exponential expansion in terms of  $O(t^3 +)$ . Continuing this expansion with the previous exponential expansion gives

$$\sum_{i=1}^N \frac{\eta_i^{k+1} \eta_i^{k+2}}{\mu_{k+1} \mu_{k+2}} + 2 \sum_{i=1}^{N-1} \sum_{j=i+1}^N \frac{\eta_i^{k+1} \eta_j^{k+1}}{\mu_{k+1}^2} = 0 \leq K \leq N-3$$

and

$$\sum_{i=1}^N \frac{\eta_i^{N-1} \sigma_i}{\mu_{N-1} \sigma} + 2 \sum_{i=1}^{N-1} \sum_{j=i+1}^N \frac{\eta_i^{N-1} \eta_j^{N-1}}{(\mu_{N-1})^2} = 1$$

Note

$$1 = \left( \sum_{i=1}^N \frac{\eta_i^{k+1}}{\mu_{k+1}} \right)^2 = \sum_{i=1}^N \left( \frac{\eta_i^{k+1}}{\mu_{k+1}} \right)^2 + 2 \sum_{i=1}^{N-1} \sum_{j=i+1}^N \frac{\eta_i^{k+1} \eta_j^{k+1}}{(\mu_{k+1})^2}$$

Taking the difference gives

$$\sum_{i=1}^N \frac{\eta_i^{k+1} \eta_i^{k+2}}{\mu_{k+1} \mu_{k+2}} = \sum_{i=1}^N \left( \frac{\eta_i^{k+1}}{\mu_{k+1}} \right)^2$$

and

$$\sum_{i=1}^N \frac{\eta_i^{N-1} \sigma_i}{\mu_{N-1} \sigma} = \sum_{i=1}^N \left( \frac{\eta_i^{N-1}}{\mu_{N-1}} \right)^2$$

These expressions are most conveniently written as vectors. Let

$$X_k = \begin{pmatrix} \frac{\eta_1^k}{\mu_k} & \frac{\eta_2^k}{\mu_k} & \dots & \frac{\eta_N^k}{\mu_k} \\ Y = \begin{pmatrix} \frac{\sigma_2}{\sigma} & \frac{\sigma_2}{\sigma} & \dots & \frac{\sigma_N}{\sigma} \end{pmatrix}$$

Then the expressions become

$$X_k * X_{k+1} = X_k * X_k \quad 1 \leq k \leq N-2$$

$$X_{N-1} * Y = X_{N-1} * X_{N-1}$$

where the \* indicates the dot product (inner product) of the vectors.

The vectors are also subject to the constraints

$$\sum_{i=1}^N X_{k,i} = 1 \quad X_{k,i} \geq 0$$

$$\sum_{i=1}^N Y_i = 1 \quad Y_i \geq 0$$

The constraints imply the vectors must all terminate on the  $n^{\text{th}}$  dimensional hyperplane  $P$  defined by the  $N$  points

$$P_i = (\delta_{1i}, \delta_{2i}, \dots, \delta_{Ni}) \quad 1 \leq i \leq N$$

where  $\delta_{ij}$  is the Kronecker delta. The vector relations may be restated as

$$X_k * (X_{k+1} - X_k) = 0 \quad 1 \leq k \leq N-2$$

$$X_{N-1} * (Y - X_{N-1}) = 0$$

Thus, either  $X_{k+1}$  is the same as  $X_k$  or the difference  $X_{k+1} - X_k$  must be perpendicular to  $X_k$ , for  $1 \leq k \leq N-2$ , and similarly, either  $X_{N-1}$  is the same as  $Y$  or  $Y - X_{N-1}$  is perpendicular to  $X_{N-1}$ . However, since each vector terminates on the plane  $P$ , their difference can be seen as a free vector lying in the plane  $P$ . Hence, either  $X_{k+1}$  must be the same as  $X_k$ , or  $X_k$  must be perpendicular to the plane  $P$ . There is only one vector perpendicular to the plane  $P$  and that is the vector

$$\perp = (1/N, 1/N, \dots, 1/N)$$

Therefore, either  $X_k = X_{k+1}$  or  $X_k = \perp$ . Thus, by the above relations the vectors can be split into two classes.

$$\begin{aligned} \exists i \rightarrow 0 \leq i \leq N-1 \ \& \ \exists \cdot \quad X_k = \perp & \quad 1 \leq k \leq i \\ X_k = Y & \quad i \leq k \leq N-1 \end{aligned}$$

Assume  $i \neq 0$ . Then

$$X_1 = X_2 = \dots = X_i = \perp$$

$$X_{i+1} = X_{i+2} = \dots = X_{N-1} = Y$$



The following relation can also be shown.

$$\begin{aligned} X_1 P_0 + X_2 P_1 + \dots + X_{n-1} P_{N-2} \\ = X_2 P_0 + X_3 P_1 + \dots + X_{n-1} P_{N-3} + Y P_{N-2} \end{aligned}$$

For ease of reading the derivation of this relation will not be given until after the proof is completed. The above relation may now be rewritten as

$$\begin{aligned} \perp P_0 + \dots + \perp P_i + Y P_{i+1} + \dots + Y P_{N-2} \\ = \perp P_0 + \dots + \perp P_{i-1} + Y P_i + \dots + Y P_{N-2} \end{aligned}$$

or

$$\perp P_i = Y P_i$$

or

$$\perp = Y$$

Therefore, unless  $Y = (1/N, 1/N, \dots, 1/N)$ , there is a contradiction and  $i = 0$ . Hence

$$X_k = Y \quad \forall k$$

or

$$\frac{\eta_i^k}{\mu_k} = \frac{\sigma_i}{\sigma}$$

But

$$\frac{\eta_i^k}{\mu_k} = \frac{\sigma_i Q_i^k}{\mu_k}$$

Therefore

$$\frac{\sigma_i Q_i^k}{\mu_k} = \frac{\sigma_i}{\sigma}$$

or

$$Q_i^k = \frac{\mu_k}{\sigma} \quad \forall i, k$$

Since the right hand side does not involve the index  $i$ , we have  $Q_i^k = Q_j^k$  for all  $i, j, k$ , thus completing the proof.

It remains to demonstrate the relation

$$\begin{aligned} X_1 P_0 + X_2 P_1 + \dots + X_{N-1} P_{N-2} \\ = X_2 P_0 + X_3 P_1 + \dots + X_{N-1} P_{N-3} + Y P_{N-2} \end{aligned}$$

The demonstration involves combining the criteria for the outputs to be Poisson and the criteria for the outputs to be independent. First the criteria for the outputs to be Poisson will be derived.

As before, define

$$F_k(t) = \Pr [L_1 > t, n(t) = k]$$

The system is then described by the system of differential equations

$$\underline{F}' = A_1 \underline{F}$$

subject to

$$\underline{F}(0) = \underline{P}$$

where  $A_1$  is the same as earlier

The solution is then known to be

$$\begin{aligned}\underline{F}(t) &= e^{A_1 t} \underline{P} = (I + A_1 t + A_1^2 \frac{t^2}{2!} + \dots) \underline{P} \\ &= \underline{P} + A_1 \underline{P} t + A_1^2 \underline{P} \frac{t^2}{2!} + \dots\end{aligned}$$

where  $A_1 \underline{P}$  and  $A_1^2 \underline{P}$  are the same as before.

Now

$$\sum_k F_k(t) = \Pr(L_1 > t)$$

and if the output is Poisson,

$$\Pr(L_1 > t) = e^{-\beta_1 t} = 1 - \beta_1 t + \beta_1^2 \frac{t^2}{2!} - \dots$$

Thus

$$\begin{aligned}\Pr(L_1 > t) &= \sum_k (\underline{P} + A_1 \underline{P} t + A_1^2 \underline{P} \frac{t^2}{2!} + \dots) \\ &= 1 - \beta_1 t + \beta_1^2 \frac{t^2}{2!} - \dots\end{aligned}$$

These results imply

$$(\sum_k A_1 \underline{P})^2 = \sum_k A_1^2 \underline{P}$$

Taking the respective sums and comparing leads to

$$\begin{aligned}C_1 \left[ \frac{\sigma_1}{\sigma} + \sum_{k=0}^{N-2} \left( \frac{\eta_1^{k+1}}{\mu_{k+1}} - \frac{\sigma_1}{\sigma} \right) P_k \right] \left[ \sum_{k=0}^{N-2} \left( \frac{\eta_1^{k+1}}{\mu_{k+1}} - \frac{\sigma_1}{\sigma} \right) P_k \right] \\ + \sum_{k=0}^{N-3} \frac{\eta_1^{k+1}}{\mu_{k+1}} \left( \frac{\sigma_1}{\sigma} - \frac{\eta_1^{k+2}}{\mu_{k+2}} \right) P_k = 0\end{aligned}$$

This is the criteria for the outputs to be Poisson.



where B is the matrix

$$B = \begin{pmatrix} -\lambda & \eta_N^1 & & & & \\ \lambda & -(\lambda + \mu_1) & \eta_N^2 & & & \\ 0 & \lambda & -(\lambda + \mu_2) & \eta_N^3 & & \\ \dots & & & & & \\ 0 & 0 & \lambda & -(\lambda + \mu_i) & \eta_N^{i+1} & \\ \dots & & & & & \\ 0 & 0 & 0 & \lambda & -(\lambda + \mu_{N-1}) & \sigma_N \\ \dots & & & & & \\ 0 & 0 & 0 & 0 & \lambda & -(\lambda + \sigma) \sigma_N \\ \dots & & & & & \end{pmatrix}$$

As before, the solution is known to be

$$\begin{aligned} \underline{G} &= e^{Bt} \underline{P} = (I + Bt + B \frac{t^2}{2!} + \dots) \underline{P} \\ &= \underline{P} + B \underline{P} t + B^2 \frac{\underline{P} t^2}{2!} + \dots \end{aligned}$$

From this result the products  $B \underline{P}$  and  $B^2 \underline{P}$  may be calculated.

In order to have independence it is necessary that

$$\Pr(L_1 > t, L_2 > t, \dots, L_{N-1} > t) = \Pr(L_1 > t) \dots \Pr(L_{N-1} > t)$$

Thus by associating the  $t^2$  terms of the joint density with the product of the densities it is seen to be necessary that

$$\sum_{k=0}^{\infty} B^2 \underline{P} = \sum_{i=1}^{N-1} \left( \sum_{k=0}^{\infty} A_i^2 \underline{P} \right) + 2 \sum_{i=1}^{N-2} \left( \sum_{k=0}^{\infty} A_i \underline{P} \right) \left[ \sum_{j=i+1}^{N-1} \left( \sum_{k=0}^{\infty} A_j \underline{P} \right) \right]$$

This leads to the requirement

$$\begin{aligned}
 C_2 & \sum_{i=1}^{N-2} \sum_{j=i+1}^{N-1} \left[ \frac{\sigma_i}{\sigma} + \sum_{k=0}^{N-2} \left( \frac{\eta_i^{k+1}}{\mu_{k+1}} - \frac{\sigma_i}{\sigma} \right) P_k \right] \left[ \sum_{k=0}^{N-2} \left( \frac{\eta_j^{k+1}}{\mu_{k+1}} - \frac{\sigma_j}{\sigma} \right) P_k \right] \\
 & + \sum_{i=1}^{N-2} \sum_{j=i+1}^{N-1} \left[ \frac{\sigma_j}{\sigma} + \sum_{k=0}^{N-2} \left( \frac{\eta_j^{k+1}}{\mu_{k+1}} - \frac{\sigma_j}{\sigma} \right) P_k \right] \left[ \sum_{k=0}^{N-2} \left( \frac{\eta_i^{k+1}}{\mu_{k+1}} - \frac{\sigma_i}{\sigma} \right) P_k \right] \\
 & + \sum_{i=1}^{N-2} \sum_{j=i+1}^{N-1} \sum_{k=0}^{N-3} \left[ \frac{\eta_i^{k+1}}{\mu_{k+1}} \left( \frac{\sigma_i}{\sigma} - \frac{\eta_i^{k+2}}{\mu_{k+2}} \right) + \frac{\eta_j^{k+1}}{\mu_{k+1}} \left( \frac{\sigma_j}{\sigma} - \frac{\eta_j^{k+2}}{\mu_{k+2}} \right) \right] P_k \\
 & = 0
 \end{aligned}$$

or with sufficient algebra  $C_1$  and  $C_2$  may be combined to yield

$$\begin{aligned}
 (13) \quad & \left[ \left( \sum_{i=1}^{N-1} \frac{\sigma_i}{\sigma} \right) + \sum_{k=0}^{N-2} \sum_{i=1}^{N-1} \left( \frac{\eta_i^{k+1}}{\mu_{k+1}} - \frac{\sigma_i}{\sigma} \right) P_k \right] \left[ \sum_{k=0}^{N-2} \sum_{j=1}^{N-1} \left( \frac{\eta_j^{k+1}}{\mu_{k+1}} - \frac{\sigma_j}{\sigma} \right) P_k \right] \\
 & + \sum_{k=0}^{N-3} \left( \sum_{j=1}^{N-1} \frac{\eta_j^{k+1}}{\mu_{k+1}} \right) \left( \sum_{i=1}^{N-1} \left( \frac{\sigma_i}{\sigma} - \frac{\eta_i^{k+2}}{\mu_{k+2}} \right) \right) P_k = 0
 \end{aligned}$$

with

$$\sum_{i=1}^{N-1} \frac{\sigma_i}{\sigma} = \frac{\sigma - \sigma_N}{\sigma} = 1 - \frac{\sigma_N}{\sigma}$$

and

$$\sum_{i=1}^{N-1} \frac{\eta_i^{k+1}}{\mu_k} = \frac{\mu_k - \eta^{k+1}_N}{\mu_k} = 1 - \frac{\eta^{k+1}_N}{\mu_k}$$

substituted in (13) and with a little manipulation it can be seen that

$$\begin{aligned}
 & \left[ \frac{\sigma_N}{\sigma} + \sum_{k=0}^{N-2} \left( \frac{\eta_N^{k+1}}{\mu_{k+1}} - \frac{\sigma_N}{\sigma} \right) P_k \right] \left[ \sum_{k=0}^{N-2} \left( \frac{\eta_N^{k+1}}{\mu_{k+1}} - \frac{\sigma_N}{\sigma} \right) P_k \right] \\
 & - \sum_{k=0}^{N-3} \frac{\eta_N^{k+1}}{\mu_{k+1}} \left( \frac{\eta_N^{k+2}}{\mu_{k+2}} - \frac{\sigma_N}{\sigma} \right) P_k \\
 & = \sum_{k=0}^{N-3} \left( \frac{\sigma_N}{\sigma} - \frac{\eta_N^{k+2}}{\mu_{k+2}} \right) P_k + \sum_{k=0}^{N-2} \left( \frac{\eta_N^{k+1}}{\mu_{k+1}} - \frac{\sigma_N}{\sigma} \right) P_k
 \end{aligned}$$

But the term on the left is simply  $C_1$  for server  $N$  implying

$$\sum_{k=0}^{N-3} \left( \frac{\sigma_N}{\sigma} - \frac{\eta_N^{k+2}}{\mu_{k+2}} \right) P_k + \sum_{k=0}^{N-2} \left( \frac{\eta_N^{k+1}}{\mu_{k+1}} - \frac{\sigma_N}{\sigma} \right) P_k = 0$$

or

$$(14) \quad \sum_{k=0}^{N-2} \frac{\eta_N^{k+1}}{\mu_{k+1}} P_k = \sum_{k=0}^{N-3} \frac{\eta_N^{k+2}}{\mu_{k+2}} P_k + \frac{\sigma_N}{\sigma} P_{N-2}$$

Since this must hold for any server, it follows

$$\sum_{k=0}^{N-2} X_{k+1} P_k = \sum_{k=0}^{N-3} X_{k+2} P_k + Y P_{N-2}$$

or

$$X_1 P_0 + X_2 P_1 + \dots + X_{N-1} P_{N-2}$$

$$= X_2 P_0 + X_3 P_1 + \dots + X_{N-1} P_{N-3} + Y P_{N-2}$$

as desired.

### VIII. Restrictions

It has been shown that only when  $Q_j^i = Q_k^i$  can the multiple servers with exponential service time distributions having different means each generate a Poisson output. It has yet to be shown how this set of equilibrium probabilities might be established. In this section the two server case will be examined in order to show a means of establishing the equilibrium probabilities and also the nature of restrictions on the dispersion of the means of the service times resulting from the necessity of establishing the equilibrium probabilities.

Consider the two server queue shown in figure two. All the appropriate assumptions are made in order to have Poisson outputs from each server. Then

$$Q_1^1 = Q_2^1$$

and since their sum must equal one,

$$Q_i^1 = \frac{1}{2}, \quad i = 1, 2$$

The following equation describes the system relative to server one, where  $R$  is the probability of assigning a customer arriving to find an empty system to server one.

$$(14) \quad (\lambda + \sigma_1) P_1 Q_1^1 = \lambda P_0 R + \sigma_2 P_2$$

The application of the equilibrium analysis given earlier in this paper shows

$$P_0 = \frac{1 - \zeta}{(1 + \zeta_1) - \zeta}$$



where

$$\xi_1 = \frac{\lambda}{\mu_1} = \frac{\lambda}{\sigma_1 Q_1^1 + \sigma_2 Q_2^1}$$

$$P_1 = \xi_1 P_0$$

$$P_2 = \xi \xi_1 P_0$$

Using these results in (14) yields

$$(15) \quad R = \frac{\sigma_1 \sigma + \lambda (\sigma_1 - \sigma_2)}{\sigma^2}$$

Thus by assigning customers which arrive to find an empty queue to server one with probability  $R$  given in (15) each server in the two server queue will have a Poisson output.

Since  $R$  is a probability, it is subject to the restriction

$$0 \leq R \leq 1$$

Therefore,

$$0 \leq \frac{\sigma_1 \sigma + \lambda (\sigma_1 - \sigma_2)}{\sigma^2} \leq 1$$

Let  $\sigma_1 = \beta \sigma$ , then  $\sigma_2 = (1 - \beta) \sigma$  and the following restrictions result.

$$(16) \quad \frac{\xi}{1 + 2\xi} \leq \beta \leq \frac{1 + \xi}{1 + 2\xi}$$

This restriction is indicated in the table below and is shown in graph form in figure three. The restriction (16) shows that in order to establish Poisson outputs from each server the dispersion of the service time means must be restricted.

$\xi$	L. B.	$\beta$	U. B.
0	0		1
.2	.143		.781
.4	.222		.777
.6	.273		.729
.8	.308		.693
1.0	.333		.667

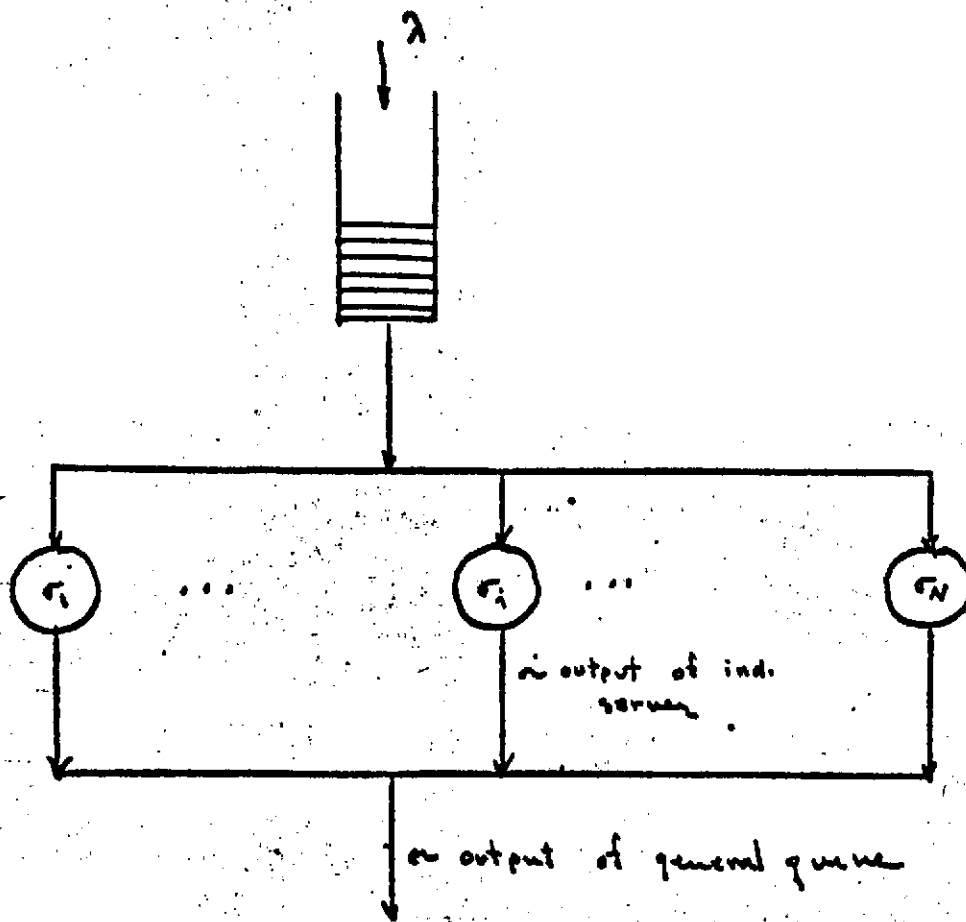


FIGURE ONE

General N Server Queue

CT

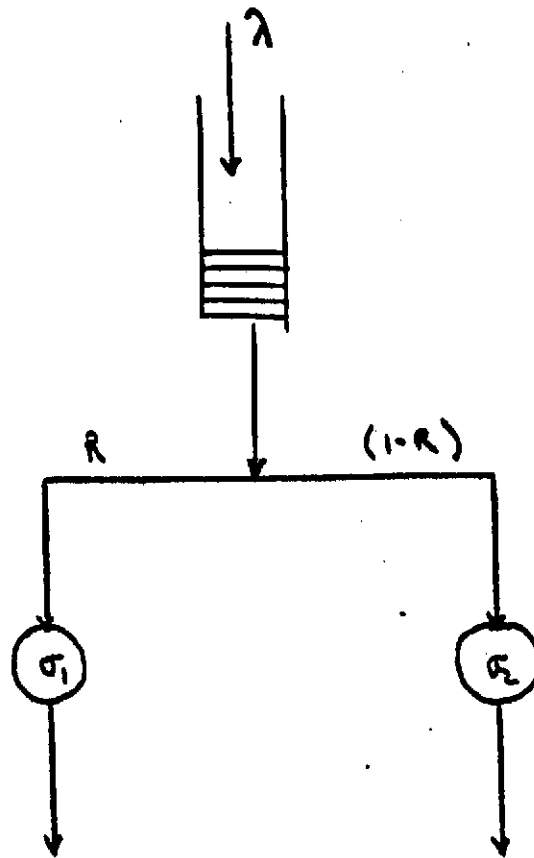


FIGURE TWO

Two Server Queue

$$\sigma_1 = \beta \sigma$$

$$\rho = \frac{\lambda}{\mu}$$

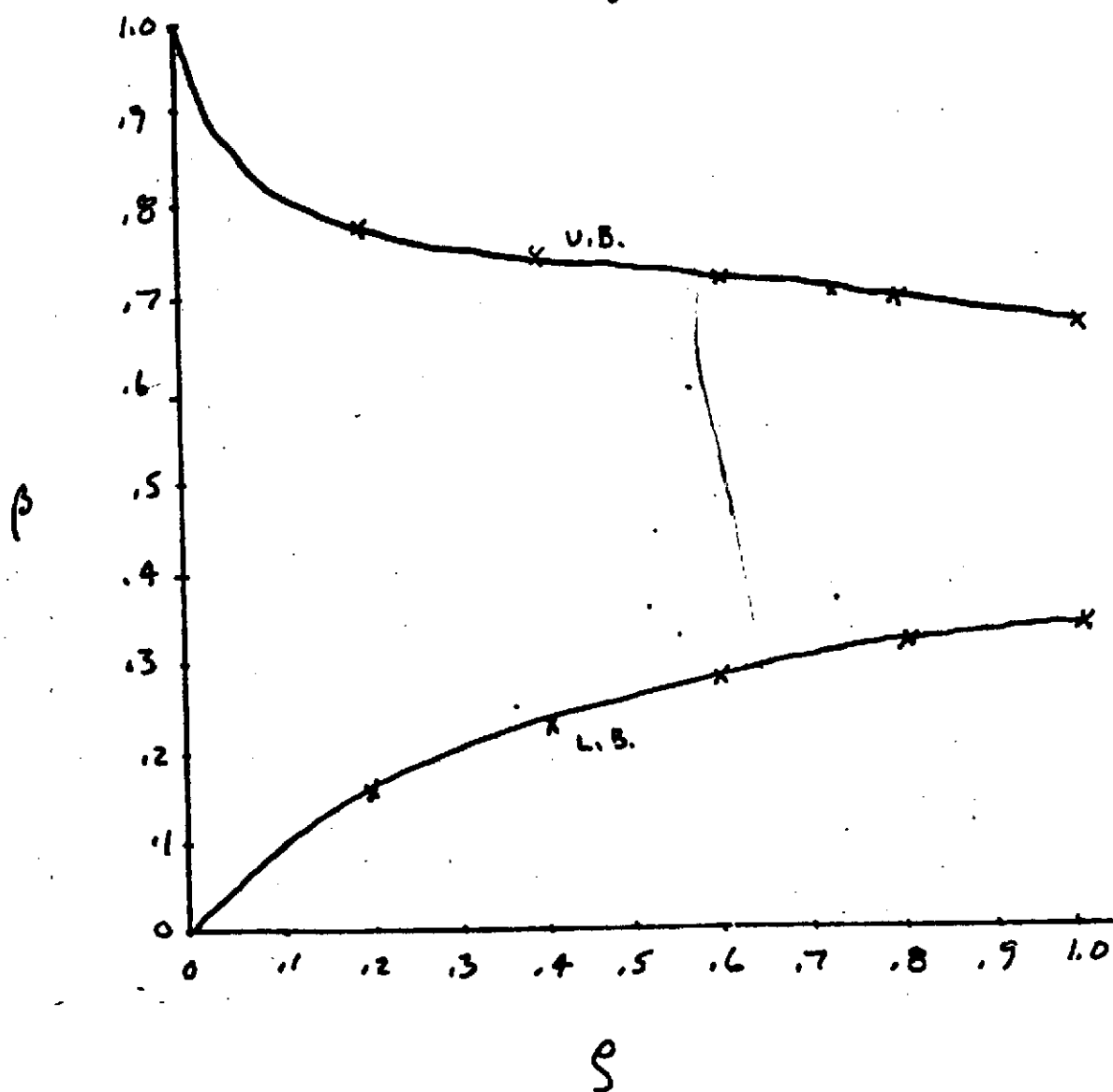


FIGURE 3

Plot of acceptable  $\beta$  versus  $\rho$  - server case

## IX. Conclusions

In this paper it has been shown that a multiple server queue, where each server has an exponential service time distribution with possibly different means, subject to a Poisson input with parameter  $\lambda$ , and after attaining equilibrium, will have a Poisson output with parameter  $\lambda$ . Furthermore, the output of each individual server  $i$  will be Poisson with parameter  $\lambda \frac{\sigma_i}{\sigma}$  if and only if it is equiprobable that a customer be in server  $i$ , conditioned on 1 customers in the system, as in server  $j$ , with the same conditioning. The two server queue was analyzed to show how such equilibrium probabilities might be established and that for such equilibrium probabilities to be possible the mean service times must not be too different. Since systems are designed by criteria other than making outputs fall into some nice statistical characterization, further investigation of the restrictions was not pursued.

REFERENCES

1. Burke, P. J., "The Output of a Queueing System", Operations Research, Vol. 4, No 6, 699-704 Dec. 1956.
2. Reich, E., "Waiting Time when Queues are in Tandem", Ann. Math. Statist. Vol. 28, 768-773, 1957.